

Xcell journal

THE AUTHORITATIVE JOURNAL FOR PROGRAMMABLE LOGIC USERS

Special Edition



**Achieving Breakthrough Performance
at the Lowest Cost**





EDITOR IN CHIEF Carlis Collins
 editor@xilinx.com
 408-879-4519

MANAGING EDITOR Forrest Couch
 forrest.couch@xilinx.com
 408-879-5270

ASSISTANT MANAGING EDITOR Charmaine Cooper Hussain

XCELL ONLINE EDITOR Tom Pyles
 tom.pyles@xilinx.com
 720-652-3883

ADVERTISING SALES Dan Teie
 1-800-493-5551

ART DIRECTOR Scott Blair



Xilinx, Inc.
 2100 Logic Drive
 San Jose, CA 95124-3400
 Phone: 408-559-7778
 FAX: 408-879-4780

© 2005 Xilinx, Inc. All rights reserved. XILINX, the Xilinx Logo, and other designated brands included herein are trademarks of Xilinx, Inc. PowerPC is a trademark of IBM, Inc. All other trademarks are the property of their respective owners.

The articles, information, and other materials included in this issue are provided solely for the convenience of our readers. Xilinx makes no warranties, express, implied, statutory, or otherwise, and accepts no liability with respect to any such articles, information, or other materials or their use, and any use thereof is solely at the risk of the user. Any person or entity using such information in any way releases and waives any claim it might have against Xilinx for any loss, damage, or expense caused thereby.

Continuing Excellence

In case you haven't heard, Xilinx recently announced its new Virtex-4™ family of FPGAs. In this special edition of *Xcell Journal*, you'll find articles devoted exclusively to Virtex-4 business viewpoints, system design challenges, engineering solutions, and engineering references.

Our "View from the Top" article is by Erich Goetting, Xilinx Vice President and General Manager of the Advanced Products Division. Erich presents an overview of the new Virtex-4 family, and gives you a guided tour of some of the new Virtex-4 technologies, as well as the inspiration and rationale behind them. Other articles in the Business Viewpoint section discuss how these new Virtex-4 devices, based on 90 nm technology, have greatly expanded high-performance processing and system integration.

You'll also find technical articles written by Xilinx marketing, applications, and development staff, as well as our partners and customers, including:

- System Design Challenges articles emphasize the Virtex-4 family advantages and leadership themes. These technical articles outline design challenges and demonstrate how the Virtex-4 solution addresses these challenges.
- Engineering Solutions articles demonstrate some of the key capabilities of Virtex-4 FPGAs and how they are used in a design. These articles provide in-depth descriptions of Virtex-4 features, IP, and tools.
- The Engineering Reference section describes some of the Virtex-4 hardware development platforms and other design solutions, to help you determine which platform is best for your application and design task.

It's Time to Re-Subscribe

This issue marks the 16th anniversary of our *Xcell Journal*. From its humble beginnings in the fourth quarter of 1988 as an eight-page, two-color newsletter, the journal has grown into an award-winning publication printed in five languages and distributed in 144 countries with a circulation of more than 60,000 readers.

Periodically, we must clean our mailing database. Beginning January 1, 2005, you must re-subscribe to continue receiving the *Xcell Journal* FREE. If you subscribed after January 1, 2005, you do not have to re-subscribe. If you subscribed before this date, please visit our site at www.xilinx.com/xcellsubscribe and take a minute to renew your FREE subscription and ensure its uninterrupted delivery.



I want to thank all of you, our readers, for your continued interest and support of the *Xcell Journal*. Please feel free to drop me a note at xcell@xilinx.com about your suggestions on how we may improve. I'd like to hear from you.



Forrest Couch

Forrest Couch
 Managing Editor

BUSINESS VIEWPOINTS

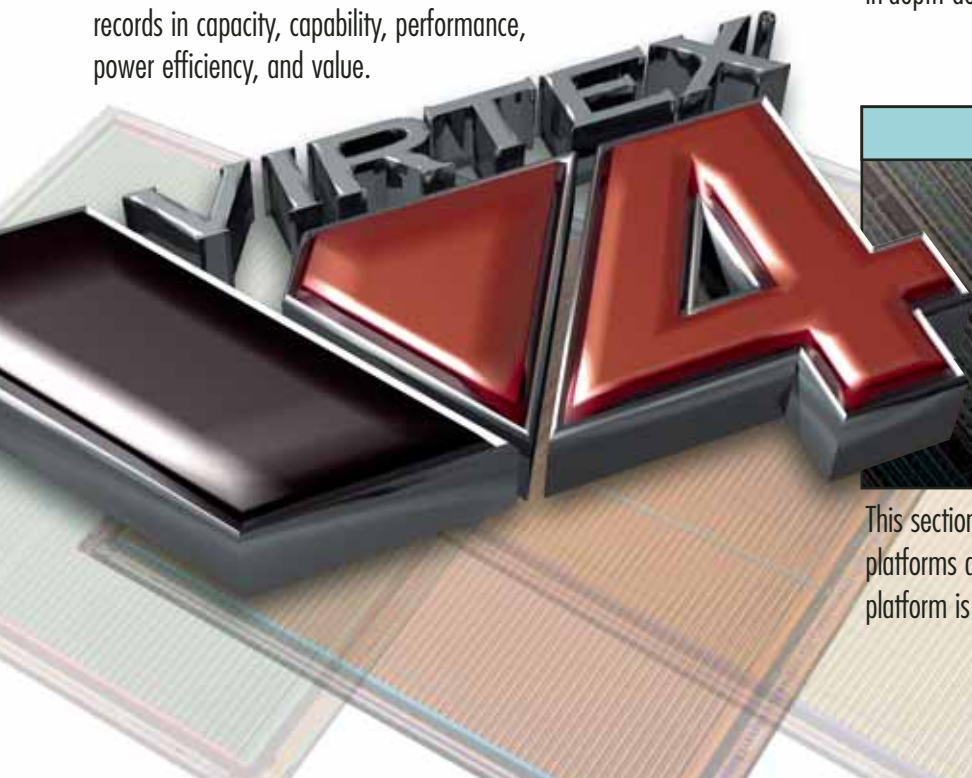


This section discusses how the new Virtex-4 devices, based on 90 nm technology, have greatly expanded high-performance processing and system integration.

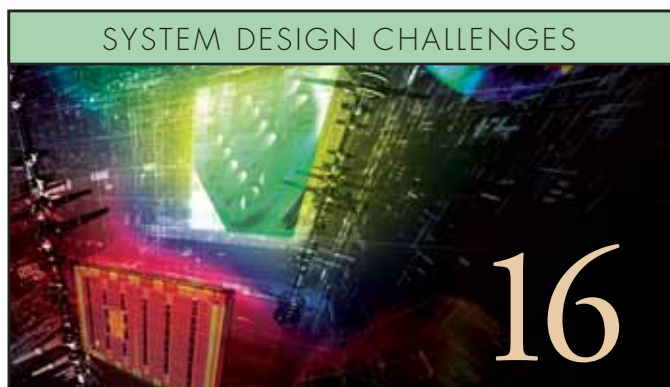
VIEW FROM THE TOP

Introducing the New Virtex-4 FPGA Family 6

The latest FPGAs from Xilinx set new records in capacity, capability, performance, power efficiency, and value.

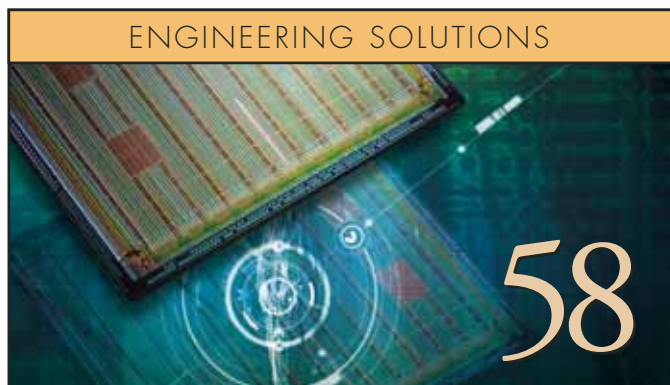


SYSTEM DESIGN CHALLENGES



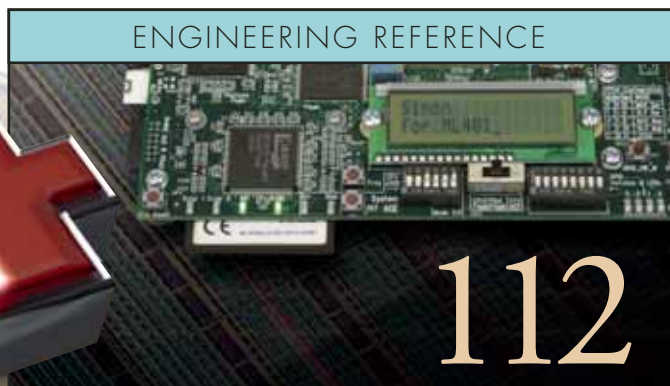
This section emphasizes the Virtex-4 family advantages and leadership themes. These articles outline design challenges and demonstrate how the Virtex-4 solution addresses these challenges.

ENGINEERING SOLUTIONS

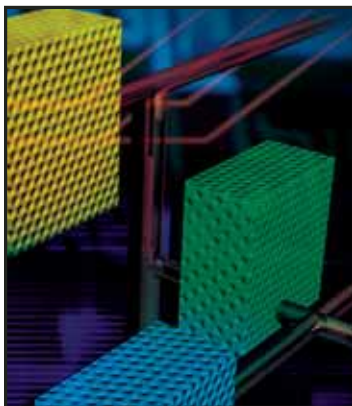
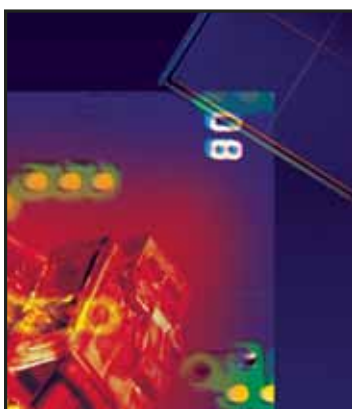
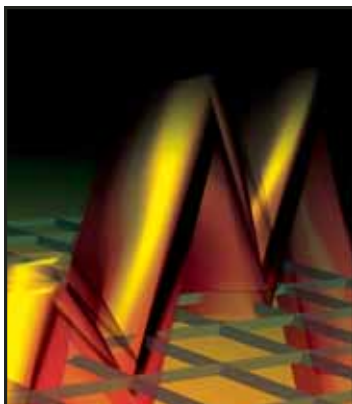


This section demonstrates some of the key capabilities of Virtex-4 FPGAs and how they are used in a design. These articles provide in-depth descriptions of Virtex-4 features, IP, and tools.

ENGINEERING REFERENCE



This section describes some of the Virtex-4 hardware development platforms and other design solutions, to help you determine which platform is best for your application and design task.



Xcell journal

View from the Top	6
-------------------------	---

BUSINESS VIEWPOINTS

Will the Evolution of Platform FPGAs Mean the End for ASICs and ASSPs?	10
EasyPath FPGAs Beat ASIC Prices	12

SYSTEM DESIGN CHALLENGES

The Virtex-4 Power Play	16
Deliver Efficient SPI-4.2 Solutions with Virtex-4 FPGAs	20
Virtex-4 Memory Interfaces	24
Designing with the Virtex-4 XtremeDSP Slice	28
Designing for Signal Integrity	32
Accelerated System Performance with APU-Enhanced Processing	36
Solving the Signal Integrity Challenge	40
Using FPGAs in Wireless Base Station Designs	42
Implementing a Cable Modem Termination System	46
Developing Next-Generation Telecommunication Networks	50
Virtex-4 FPGAs for Software Defined Radio	54
Virtex-4 FPGAs in Rugged LCD Monitors	56

ENGINEERING SOLUTIONS

ISE 6.3 Software — Unleash the Power of Virtex-4 FPGAs	58
FIFOs Made Easy	61
Digital Clock Management	64
Virtex-4 Clocking Resources	66
Alpha Blending Two Data Streams Using a DSP48 Technique	68
Dynamic Phase Alignment with ChipSync Technology	72
Lock Your Design with the Virtex-4 Security Solution	75
Dynamic Reconfiguration of Functional Blocks	78
Designing with Virtex-4 Embedded Tri-Mode Ethernet MAC	80
Emerging Design Methodologies Elicit the Power of Virtex-4 FPGAs	84
Integrate EDK-Created Embedded Processor Subsystems	88
Optimizing Virtex-4 High-Performance Designs	92
Selecting Connectors for Multi-Gigabit Transceiver Designs	96
Xilinx/Micron Partner to Provide High-Speed Memory Interfaces	100
Harvesting the Flexibility of Virtex-4 RocketIO Transceivers	102
Optimize Memory Subsystem Performance with Network FCRAM	105

GENERAL

Using Spartan-3 FPGAs to Implement High-Performance DSP	108
---	-----

ENGINEERING REFERENCES

Virtex-4 ML401 Evaluation Platform	112
Virtex-4 FPGA Source-Synchronous Interfaces Design Kit	113
Virtex-4 ML461 Advanced Memory Development System	114
Memec Virtex-4 Board Solutions	115
Avnet Virtex-4 Evaluation Kits	116
Nu Horizons Virtex-4 Development Platform	117
TED DDR2 Memory Evaluation Board	118

Introducing the New Virtex-4 FPGA Family

The latest FPGAs from Xilinx set new records in capacity, capability, performance, power efficiency, and value.



by Erich Goetting
Vice President &
General Manager,
Advanced Products Division
Xilinx, Inc.
erich.goetting@xilinx.com

Welcome to the Xilinx® Virtex-4™ edition of the *Xcell Journal*. We've created this special issue to show you the new Virtex-4 FPGA family, and how its innovations enable the creation of next-generation systems that do more than ever thought possible only a few years ago.

In this article, I'll take you behind the scenes for a guided tour of some of the new technologies, as well as a bit of the inspiration and rationale behind them.

With more than 100 innovations, the Virtex-4 family represents a new milestone in the evolution of FPGA technology. After conducting extensive interviews with leading design engineers worldwide, we knew that they wanted the following things in an advanced next-generation FPGA family:

- Higher performance
- Higher logic density
- Lower power
- Lower cost
- More advanced capabilities

At the heart of the Virtex-4 FPGA is our next-generation 90 nm triple-oxide 10-layer copper CMOS process technology.

It's relatively easy to deliver on one or two of these items – our challenge was to deliver all of them at the same time. We did this through a combination of innovative process and circuit design, process development, the ASMBL architectural approach, and the use of advanced embedded functions.

Development work on the Virtex-4 family (code-named “Whitney” after the highest mountain in the continental United States) began more than two years ago. It represents the creativity and dedication of hundreds of engineers, spanning integrated circuit design and layout, software and IP development, process development, testing and characterization, systems and applications engineering, technical documentation, and product marketing.

One of the most remarkable developments embodied in the new Virtex-4 FPGA family is the ASMBL architecture, which represents a fundamentally new way of constructing the FPGA floor plan and its interconnect to the package. First of all, ASMBL enables I/O pins, clock pins, and power and ground pins to be located anywhere on the silicon chip, not just along the periphery as with previous approaches. This in turn allows power and ground pins to be brought directly into the center of the silicon die, thereby significantly reducing on-chip IR drops that can occur with the largest FPGAs running at the highest frequencies.

Clock input pins are also located in the center of the die, which reduces clock latency. This is because clock networks need to have equal delay to all endpoints (that is, minimum skew), and thus the clock must emanate from the center. In periphery-connected clock input pins, the signal first traverses from the edge of the die to the center, and is then distributed to all regions. The Virtex-4 ASMBL design eliminates this traversal completely, and thus directly reduces the clock network propagation delay.

In addition to its electrical advantages, ASMBL provides another significant benefit in that it allows a more flexible – and thus more precise – allocation of on-chip resources.

That in turn has enabled us to offer Virtex-4 devices in three unique platforms, each with a different mix of on-chip resources:

- The LX platform, optimized for logic applications
- The SX platform, optimized for high-end DSP applications
- The FX platform, optimized for embedded processing and high-speed serial applications

A Look Inside the Virtex-4 FPGA

At the heart of the Virtex-4 FPGA is our next-generation 90 nm triple-oxide 10-layer copper CMOS process technology. While that's quite a lot of adjectives, every one of them is incredibly important. The first, 90 nm, refers to the “drawn” gate length of the smallest transistors. As transistors get smaller, they get faster, use less dynamic power, and enable higher complexity at lower price points. Chip designers think in terms of “transistor budgets,” which are now in the billion transistor range.

Triple-Oxide 90 nm CMOS Technology

Triple-oxide technology refers to the number of transistor oxide thicknesses available in the process. More oxide thicknesses allow more tuning of performance and power in the device circuitry, and enable Virtex-4 devices to deliver industry-leading performance while dramatically lowering power consumption.

One of our key inputs from many engineers was that performance and power were very important constraints in their systems designs, and that they needed both high performance and low power. With a dual-oxide 90 nm process, we would have had to choose performance or power. This wasn't good enough. By employing a triple-oxide 90 nm process, we achieved high performance and low power.

The 10-layer copper refers to the number of metal interconnect layers and their

material, which is copper rather than aluminum (the traditional material). More layers provide more routing in less space and shorter connection distances. Copper reduces resistance compared to aluminum, and thus speeds signal interconnect and reduces on-chip power-distribution IR drop. As clock rates go up and voltages go down, these considerations have become increasingly important, and have driven the industry-wide shift to copper interconnect.

The Virtex-4 logic fabric was completely re-engineered to fully take advantage of the 90 nm triple-oxide CMOS process, resulting in the highest performance fabric ever, with system clock rates in excess of 500 MHz (at three LUT levels). At the same time, static power was cut in half compared to 130 nm Virtex-II Pro™ devices, as was dynamic power.

Thus, while some industry pundits were proclaiming that the future of deep submicron CMOS devices was getting hotter and hotter, with chip temperatures destined to reach that of rocket nozzles and the surface of the sun, the Virtex-4 design's creative approach has turned that conventional wisdom on its head, resulting in overall power reductions of 50% compared to our previous 130 nm generation. In many applications, such as DSP functions, power levels are reduced even more – as much as 90%. No wonder design engineers say that Virtex-4 FPGAs are cool – they literally are.

High-Performance Clocking

Clocks were rated as one of the most important and critical FPGA resources in our surveys of design engineers. Quantity, quality, connectivity, frequency, duty cycle, jitter, and skew all made a big difference.

To take clocking to the next level in Virtex-4 devices, all global clock resources were made fully differential, thereby reducing skew, jitter, and duty-cycle distortion. This marks the first implementation of differential clocking in a programmable logic device. Not only that, but the number of global clocks was increased to 32, for every

device, and internal connectivity options enhanced to allow any region to use any 8 clocks simultaneously.

500 MHz Synchronous Memories and FIFOs

On-chip synchronous block RAM was enhanced to run at 500 MHz. Built-in support for first-in first-out (FIFO) memories was included directly in the block RAM unit, enabling the same 500 MHz operation for FIFOs (approximately a 2X speedup over fabric-based FIFOs), while eliminating the need for any additional logic cells or complex FIFO designs.

If you're designing systems requiring ECC (error checking and correcting) memory, Virtex-4 devices have built-in ECC support, with single-bit correct and double-bit detect. ECC is common in infrastructure equipment in networking, telecom, storage, servers, instrumentation, and aerospace applications, and provides the highest levels of data integrity. Like the integrated FIFO support, the integrated ECC eliminates the cost and delay of fabric-based solutions.

Speaking of on-chip memory, Virtex-4 devices continue to offer SelectRAM™ memory, whereby each LUT is transformed into a 16 x 1 RAM, ideally suited for building high-speed register files and local buffers.

At the other end of the spectrum, interfaces to external memory devices such as DDR, DDR2, QDR-II, and RLDRAM-II are dramatically enhanced through our new ChipSync™ technology, which offers memory interface speeds at rates limited only by the speed of the external memory devices.

The new Virtex-4 ML461 Advanced Memory Development System contains fully functional and hardware-proven reference designs for all of today's most popular memory technologies. If you plan to use external memory, I highly recommend that you check this out.

DSP Performance of 256 GigaMAC/s

In the DSP domain, we incorporated some of the world's fastest multiply accumulate (MAC) technology. The XtremeDSP™ slice can perform an 18 x 18 signed multiply and 48-bit accumulate every 2 ns.

The Virtex-4 LX, FX, and SX platforms include the breakthrough XtremeDSP technology. With the new SX platform we did something completely new – we dramatically increased the ratio of DSP units to logic cells. Given the highly integrated nature of XtremeDSP slices, they need only small amounts of logic fabric to implement most common DSP functions, and thus increasing the ratio provides a significant increase in DSP compute power per unit silicon area. In fact, SX devices provide a 10X performance increase per unit cost over previous solutions.

Power is dramatically reduced as well, with more than a 10X reduction for multiply/add functions from previous FPGA solutions. The Virtex-4 SX55 contains 512 XtremeDSP slices, providing an aggregate DSP compute performance of 256 GigaMAC/s, making it one of the most powerful DSP devices ever manufactured.

The state-of-the-art XtremeDSP slice employs new “silicon algorithms” developed by a company called Arithmatica™. Many different architectures exist for implementing multiplication, and the Arithmatica architecture is truly a breakthrough. We are excited to see it available for the first time to FPGA users. For more information, visit Arithmatica's website at www.arithmatica.com.

The Evolution of Advanced I/O Technology

I/O continues to be a critical success factor for today's systems designers. During the last decade, we have seen four major changes in I/O. First was the shift away from 5V, the result of the need to scale voltages as we scaled the transistor. This in turn led to the plethora of I/O standards that we are all familiar with today: SSTL, HSTL, LVDS, and LVCMOS 1.5. The Virtex-4 SelectIO™ resource continues to lead the industry, supporting virtually every I/O standard in use today on every pin.

XCITE On-Chip Termination

The second major change was the transition from lumped loads to transmission line loads – again the direct result of Moore's Law. As transistors got faster and clock rates increased, I/O edge rates

increased as well. But because the propagation speed of signals is a constant, dictated by the speed of light, we entered the realm in which a signal on one end of a wire was no longer the same as the signal on the other end of the same wire. This is what transmission lines are all about, and their appearance during the last few years has driven a sea change in all aspects of signal interconnect and I/O design.

To make sure that these signal “waves” don't start “splashing” uncontrollably, transmission lines need to be driven, built, and received using proper signal integrity approaches, the most critical of which is termination. Traditionally implemented with discrete resistors on the PCB, termination layouts can become exceedingly difficult around high-density pinouts like those used in FPGAs. This often dictates more PCB layers and thus more system cost.

Virtex-4 FPGAs include our third-generation of XCITE™ integrated digitally controlled termination technology. Offering a precisely controlled source impedance at the output drive pin, it is designed to enable the driving of transmission lines without external components, with maximum speed and signal integrity, and with straightforward PCB layout and layer stack-ups.

Likewise, on inputs, XCITE offers parallel termination for single-ended inputs and true differential termination for differential inputs. Termination occurs on the end of the transmission line at the die, not on the way there on the PCB, offering maximum signal integrity. Many customers report that the XCITE technology has saved them many PCB layers, increased PCB packing density, and saved them substantial dollars in their bill of materials.

Source-Synchronous Interfaces

The third major change was the shift from system-synchronous to source-synchronous interfaces. Traditional system-synchronous interfaces work by distributing a single clock to all transmitters and receivers in the system, and transmitting data between source and destination within a single clock cycle. This makes the data rate inversely proportional to the sum of clock-

to-out, transmission line delay, and input setup time.

Typically, system synchronous interfaces top out at speeds in the range of 100 MHz. To go faster, source-synchronous interfaces transmit a clock along with the data, and the receiver uses this clock to capture the data. Using this technique, along with double-data-rate transmissions, enables parallel I/O data rates in excess of 1 Gbps.

The challenge of source-synchronous interfaces is that each interface generates a new clock domain at the receiver. On top of this, to operate at high speeds, the precise alignment of clock and data at the receiver is paramount. To address this new world of source-synchronous interfaces, Virtex-4 devices include the breakthrough ChipSync technology. ChipSync units lie between the SelectIO technology and the core FPGA fabric, are available on every I/O pin on the device, and serve to transmit and receive high-speed source-synchronous data and clocks, achieving speeds of 1 Gbps per pin pair.

On the receiver, precise digital delay lines work internally to align data signals to each other, and then to align these to the received clock. The captured data is synchronized and transferred to the selected FPGA core clock domain.

To operate at maximum data rates, the transmit and receive units include parallel-to-serial and serial-to-parallel conversion units, respectively. Using ChipSync technology is virtually automatic for most designs, as it is utilized automatically in the various Xilinx IP cores and reference designs.

Networking interfaces such as SPI-4.2 and HyperTransport™, and memory interfaces such as DDR, DDR2 SDRAM, and QDR II SRAM, all employ the Virtex-4 ChipSync technology. And if you're designing your own source-synchronous interface, the ChipSync wizard gives you complete control and an easy-to-use GUI that lets you dial in exactly what you want to build.

Multi-Gigabit Serial Interfaces

The fourth major change in I/O has been the rapid adoption of high-speed serial interfaces. For years, serial interfaces were limited to long-distance communications,

such as those used in fiber-optic links in the SONET/SDH world and the Ethernet links like 100BASE-T.

A key breakthrough occurred in the late 1990s, in which high-speed serial transceivers (which traditionally had been designed using complex process technology such as GaAs [Gallium-Arsenide]) were for the first time created using advanced design techniques using standard CMOS. Once implemented in CMOS, these transceivers had lower cost and much lower power, and could even be integrated into complex CMOS chips.

Virtually overnight, gigabit serial technology changed from a rare, expensive, and power-hungry technology to a common, low-cost, and very power-efficient technology. This has been the economic and technical impetus behind the industry's "Serial Tsunami," in which interface after interface has shifted from parallel to gigabit serial links. Two common examples are visible in today's computer architectures, with the shift from parallel PCI to 2.5 Gbps serial PCI-Express™, and the shift from the parallel ATA drive interface to the Serial ATA interface.

There are more than a dozen multi-gigabit serial interfaces in widespread use today, with more being introduced every year. The Virtex-4 FX family provides our third-generation RocketIO™ multi-gigabit serial transceiver technology. Spanning speeds from 622 Mbps to more than 10 Gbps, each Virtex-4 RocketIO transceiver is programmable and can implement a myriad of speeds and serial standards. Link-layer IP is available for such standards as PCI Express, Serial-ATA, FibreChannel, Gigabit Ethernet, and Aurora, to name a few.

In addition, Virtex-4 FX devices each include multiple embedded tri-mode (or 10/100/1000) Ethernet MACs, making implementation of compliant Ethernet devices simpler and faster than ever.

Application-Specific Embedded Processing

Virtex-4 embedded processing solutions include full support for both MicroBlaze™ 32-bit soft CPUs on all devices, and embedded PowerPC™ 32-bit RISC CPUs on all Virtex-4 FX devices. The versatile MicroBlaze soft CPU runs at clock rates

over 165 MHz on Virtex-4 devices, and delivers more than 140 DMIPS.

The number of CPUs in one device is limited only by your imagination, and of course by the available logic cells. The powerful PowerPC CPU runs at clock rates up to 450 MHz and delivers up to 702 DMIPS each. The first PowerPC processor available by any manufacturer on 90 nm, the PowerPC processor is incredibly power-efficient, using only 29 mw/DMIPS. This makes it among the lowest power microprocessors available from any manufacturer worldwide.

New Auxiliary Processing Unit (APU) technology connects the CPU to the FPGA fabric, enabling implementation of acceleration hardware for virtually any application. Once only the domain of high-budget ASIC and ASSP design teams, the Virtex-4 FPGA's architectural ability to combine application-specific hardware acceleration with high-performance RISC CPUs shatters traditional barriers of cost, time-to-market, and risk.

During the next few years, I expect to see more and more instances of application-specific acceleration, as it truly offers the ability to deliver very high performance at low cost and low power. A recent research program completed within Xilinx Research Labs, led by Dr. Kees Vissers, demonstrated a 20-fold speedup for an encryption/decryption (IPSEC) application over the base PowerPC processor. Using only 135 mW, it outperforms a 3.2 GHz Pentium™-4, while at the same time reducing power by 99%. That, in my opinion, is what state-of-the-art embedded processing is all about.

Conclusion

I hope that you've enjoyed reading a bit about the Virtex-4 Platform FPGA and the factors that drove its design. From the breakthrough ASMBL architecture and the triple-oxide 90 nm CMOS process technology, to the world's most capable embedded processing and multi-gigabit serial solutions, Virtex-4 devices offer an unparalleled set of enabling technologies for your next-generation systems designs. I look forward to seeing the creativity of the world's designers in tomorrow's products. 🌟



Will the Evolution of Platform FPGAs Mean the End for ASICs and ASSPs?

Today's multi-platform FPGAs shake up ASIC/ASSP suppliers.

by Richard Sevcik

Executive Vice President, Programmable Logic Systems and Intellectual Property/Cores and Software Solutions Groups
Xilinx, Inc.

publicrelations@xilinx.com

The debate over FPGAs as a viable alternative to ASICs and ASSPs has been ongoing for nearly a decade. Industry analysts iSupply, Gartner Dataquest™, and others have documented the trend in decreasing ASIC design starts and the increase in FPGA design starts.

Next-generation platform FPGA devices based on 90 nm have greatly expanded high-performance processing and system integration options. They continue to push ASIC design starts lower as additional application solutions are defined.

With the beginning of the new millennium, the debate continued with the introduction of Xilinx® Virtex-II™ and Virtex-II Pro™ devices – the industry's first platform FPGAs. These high-performance devices, with their flexible device integration capability, programmable I/O, and significantly lower overall design cost, helped to usher in and establish SoC design methodology and quickly assumed innumerable ASIC SoC designs.





The addition of high-performance RISC CPUs, block RAM, multi-gigabit high-speed serial I/Os, dedicated DSP functions, and other system enhancements introduced technological advances that further solidified the rise of platform FPGAs over their ASIC SoC counterparts. However, to get high-performance DSP, processing, or connectivity features for a specific applications domain, designers were typically forced to purchase the largest, costliest devices. The larger parts had the biggest helpings of advanced features, while the smaller parts had reduced portions of the same.

Today, a new breed of domain-optimized, multi-platform FPGAs from Xilinx – the Virtex-4™ family – promises multi-dimensional application scaling based on required features and cost goals. By combining the economic benefits of an innovative columnar architectural approach with advances in process technology (90 nm/300 mm), Xilinx is poised to move beyond the \$5.1 billion programmable logic market to capture additional share in the \$84 billion ASIC and ASSP markets (Source: Gartner Dataquest 2007).

Just the Right Mix

Based on the revolutionary Advanced Silicon Modular Block (ASMBL) columnar architectural approach, Xilinx can now cost-effectively develop multiple FPGA platforms, each with different combinations of feature sets. Thus, a specific platform can be optimized specifically for a certain domain of applications – such as logic, DSP, connectivity, and embedded processing – to meet application requirements previously delivered only by ASICs, ASSPs, and similar devices, while remaining programmable at heart.

Not only does the designer or design team have a choice in selecting the ideal platform, they also have a choice in choosing the device size with just the right feature mix to best achieve needed capability and performance at the lowest possible cost.

This unique flexibility and ability to create optimal application domain subsystems sets even higher standards for FPGAs. Devices that are both hardware- and

software-programmable enable more flexible implementation options than either ASIC or ASSP devices. Reinvestigating, changing, or enhancing system architecture at any time in the development process provides the ultimate tool kit to meet application requirements.

Designers can use this same capability to evolve hardware in the field to meet new

technology is used throughout the world. No two people use the same technology, systems, or software, nor do they subscribe to or want the same content.

Higher costs and longer design times for ASICs and ASSPs relegate their primary uses to proven lower-risk, very-high-volume applications. The rapid and significant increase in ASIC development costs clearly



requirements or avoid expensive hardware upgrades. This flexibility becomes paramount given today's many emerging and competing standards.

The "Total Cost" Advantage

FPGAs have demonstrated a clear and consistent trend in reducing cost and making FPGA technology more suitable for a wider range of applications. The combination of 90 nm silicon fabrication technology with 300 mm wafers results in a cumulative effect: increasing the number die-per-wafer five times over previous devices. Increasing the die-per-wafer together with architectural integration enables substantially lower system costs.

A key and often overlooked component in favor of programmable logic's economic advantage is clearly demonstrated in how

gives the advantage to platform FPGAs in today's leading-edge applications. The overall cost benefit of zero NRE pushes the high-volume ASIC or ASSP crossover point upwards, locking in FPGAs like never before.

Conclusion

Domain-optimized multi-platform FPGAs are revolutionary in their ability to accelerate the deployment of FPGA technology into many more application areas. The combined leverages of reduced risk, dramatically shorter design cycles, and zero NRE will soon move all but the highest volume applications away from cell-based ASIC implementation toward more flexible, forgiving architectures like today's domain-optimized FPGAs. For more information, visit www.xilinx.com/virtex4/.

EasyPath FPGAs Beat ASIC Prices

EasyPath is the most comprehensive volume production solution in the industry.

by Gokul Krishnan

Sr. Marketing Manager, Market Specific Products Group
Xilinx, Inc.

gokul.krishnan@xilinx.com

Balaji Thirumalai

Sr. Marketing Manager, Worldwide Marketing
Xilinx, Inc.

balaji.thirumalai@xilinx.com

The risk of deploying ASIC solutions has worsened in magnitude with the move to smaller process geometries. As design complexity increases, customers are looking for a viable solution that offers low design, unit, and total costs, high-level system integration, design flexibility, easy-to-use design tools, a rich selection of IP, and fastest time to market.

Customers are increasingly turning to other alternatives to avoid the pitfalls of ASICs – high NRE and re-spin expenses, slow turnaround times, complex design environments, and hidden conversion, verification, and development costs. In this article, we'll analyze two such alternatives: Xilinx® EasyPath™ FPGAs and structured ASICs.

Structured ASIC product offerings tend to be similar to FPGAs in that they have predefined combinations of gates, memory, and I/Os. However, their architectures tend to trade off flexibility in favor of reduced area to achieve their cost targets. The reality remains that a vast majority of designs intended for ASICs are originally prototyped in an FPGA, yet there are still problems with FPGA-to-structured-ASIC conversions. EasyPath FPGAs offer the best migration path to high-volume production at the lowest cost possible.

EasyPath FPGAs

EasyPath FPGAs are the industry's only customer-specific and flexible solution for volume production priced lower than structured ASICs.

EasyPath FPGAs are identical to our standard FPGA offerings but use patented testing techniques and customer-specific test patterns to significantly improve FPGA yields. You can reap the benefits of these improved yields in the form of lower costs, because Xilinx only tests those parts of an

FPGA that are actually used in your design.

With EasyPath FPGAs, you can realize a 30-80% reduction in prices when you move to high volume, as compared to standard FPGAs. EasyPath FPGAs are available across six platforms, four different product families, and 28 different devices over a range of gate and memory counts.

EasyPath FPGAs are identical to their standard FPGA counterparts, effectively eliminating any conversion work. Once you have frozen your design, Xilinx can deliver EasyPath parts in high volume in eight weeks. This compares favorably against structured ASIC companies, which typically take 12-14 weeks from prototype signoff to production.

Structured ASICs

Structured ASICs are a variant of the gate arrays of yesteryear, but they use a "sea of modules" approach as opposed to a "sea of gates" approach. The architecture of each module varies depending on the vendor, but in general is some combination of NAND gates, inverters, flip-flops, and muxes.

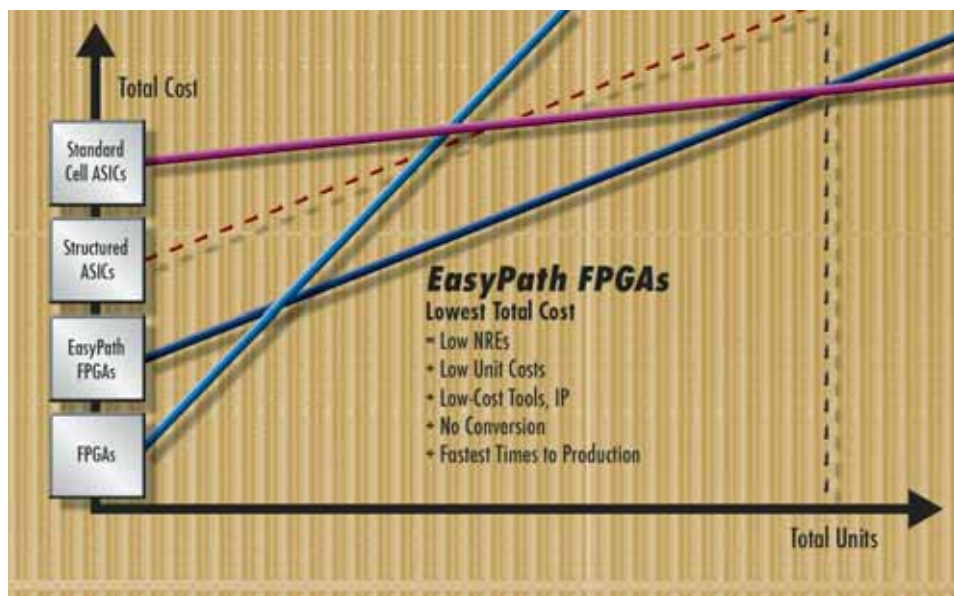


Figure 1 – EasyPath FPGAs offer the lowest total cost solution.

Structured ASICs promise cost savings primarily as a result of customizing fewer mask layers per design, unlike standard cell ASICs that use all-custom metal layers. Structured ASICs use only the top few (typically two to four) metal layers; the base modules are all buried in the lower layers, with their ports coming up to the programmable layers. During the fabrication phase, the connections between various ports are made to realize the requisite logic.

The Lowest Total Cost Solution

Figure 1 shows the comparative economics of standard cell ASICs, structured ASICs, FPGAs, and EasyPath FPGAs. FPGAs have traditionally offered a zero-NRE solution, which has led to their broad adoption. Standard cell ASICs have a high NRE and a relatively low unit cost, but with the overhead discussed earlier. Structured ASICs promise to lower the NRE at a unit cost that is higher than that of standard cell ASICs, but lower than that of standard FPGAs.

With next-generation EasyPath FPGAs, you can now enjoy unit prices as well as NREs that are lower than structured ASICs. The combination of the industry's lowest NRE charges (starting at \$75K); low cost design tools and IP; prices below structured ASICs; fastest times to production; and no hidden conversion charges show how EasyPath

FPGAs are the industry's lowest total cost solution for volume production.

Unmatched Choice of Platforms

Structured ASIC vendors can roughly be grouped into two camps based on their ability to address IP-centric designs. On the one hand are those that have a wide portfolio of IP; on the other are companies that typically can only address generic designs. With the recent announcement of next-generation EasyPath FPGAs from Xilinx, both of these segments can be addressed economically and efficiently.

Xilinx now offers four families and six platforms, with 28 devices from which to choose. This comes with all the benefits of the FPGA ecosystem that Xilinx customers are already used to – hard IP such as the IBM™ PowerPC™, MGTs, and XtremeDSP™ blocks, as well as 600+ proven soft IP cores and low-cost design tools.

Some structured ASIC vendors focus exclusively on generic designs or logic-heavy designs. This class of design tends to be very price competitive. Xilinx is now able to offer a more compelling solution than any structured ASIC vendor with its Spartan-3™ EasyPath FPGAs, which are priced below these structured ASICs.

For designs that require a lot of IP and system integration such as PowerPC processors, DSP, high-speed I/O, or Ethernet

MACs, translation to a structured ASIC vendor often requires a re-validation of the IP on the vendor's silicon platform of choice. With Xilinx Virtex-4™ EasyPath solutions, you get the same wide range of validated IP as with standard FPGAs. There is no additional fee required to migrate the IP to a volume solution.

The bottom line is that whether it is a generic design or an IP-centric design, EasyPath FPGAs offer very competitive and cost-effective solutions for high-volume migration when compared to structured ASICs, all from a single trusted supplier. Migration to structured ASICs, on the other hand, can pose a number of challenges.

Conversion-Free Methodology

The vast majority of IC design starts begin with FPGA prototyping, followed by a conversion to a volume solution. This carries the inherent risk of redesigning and re-verifying the design in the target architecture, along with the related costs of re-spins, conversions, and a host of other design issues. The conversion from FPGA to structured ASIC is not seamless; rather, it is fraught with risks.

One issue faced by structured ASIC companies revolves around the mapping of memories from an FPGA to a structured ASIC. FPGAs generally tend to have columnar memory architectures and offer an efficient means to form larger memory structures when required. On the other hand, the use of distributed memory blocks in some structured ASIC architectures can pose problems when large contiguous blocks are required by the design.

The need to join together blocks that are physically separated to form a larger block that is logically monolithic can increase routing congestion. This can not only potentially deteriorate the access times of those memory structures but also leave fewer routing resources available for logic, thus impacting design performance.

With EasyPath FPGAs, there is no conversion. EasyPath FPGAs are exactly the same as the standard FPGAs on which a design is prototyped – the only difference is that the latter are completely programma-

ble, while the former are not. As a result, memory mapping and performance achieved in an EasyPath FPGA is identical to that achieved in a standard FPGA.

Another problem that some structured ASIC companies face has to do with pad limitations. It is fairly well known that as process nodes shrink, more and more designs become pad-limited in ASICs. To get an adequate number of pads, structured ASIC vendors sometimes have to grow their die size and increase the effective cost to end customers. This problem is compounded by the fact that structured ASIC I/Os tend not to be as flexible as FPGA I/Os.

To keep I/O structures small and less area-intensive, structured ASIC vendors have to make some difficult choices about what standards they want to address and how. In cases where designs require large buses of input and output I/Os (for example, SSTL2 buses for SDRAM, or HSTL buses for certain telecom protocols), the limitations in the design of I/O structures can make it difficult to achieve pin compatibility in the FPGA-to-ASIC conversion. The end result is that customers have to either re-spin their board or migrate to a larger device – both unpalatable options. None of these are issues with EasyPath FPGAs because of the one-to-one mapping between them and standard FPGAs.

Apart from memory and I/Os, there is a whole other host of issues, including difficulties with IP translation and testing, when moving from FPGAs to structured ASICs. FPGA cost reduction plans that involve converting to structured ASICs in order to get a smaller die are likely to trigger design changes and schedule risks.

The EasyPath solution, on the other hand, is neither an ASIC conversion nor a mask-programmed FPGA. No conversion or silicon differences are involved, so there

are no long lead times, no timing or pinout changes, no need for product qualification, no lost feature support, and no risk of a design failure. In addition to eliminating any hidden design or qualification expenses and the risks of ASIC conversions, EasyPath FPGAs are delivered in eight weeks in production volume, allowing you the benefits of faster time to market or more time to perfect your designs

Unprecedented Flexibility

One of the major advantages of FPGAs over ASICs is the flexibility to make design changes in case of a specification change or a design error. Traditionally, customers have had to forgo this advantage as they move from FPGAs to an

now enjoy greater flexibility while also saving on BOM and inventory costs. For example, you can use one bitstream to perform system diagnostics on the entire system and then load the second application-specific bitstream. This reduces associated manufacturing system costs.

Xilinx offers EasyPath FPGA devices with LUTs and I/Os tested for drive strengths and slew rates, allowing revisions like engineering change orders at the LUT or I/O level. In many instances, even after the customer design is fully functional and certified, flexibility with I/O drive strengths and slew rates is critical.

For instance, a line card in a router might need to have the drive strength (and slew rate) adjusted a notch or two depending on what load it sees. EasyPath customers can choose to have a range of drive strengths available to them for certain I/Os. The unique flexibility is implemented on an as-needed basis. This eliminates any re-spin and conversion-related engineering effort, delay, and expenses associated with ASICs and structured ASICs.

Conclusion

EasyPath FPGAs from Xilinx offer a seamless one-for-one, no-conversion volume reduction solution across an industry-leading portfolio of product families. The comparison between EasyPath FPGAs and structured ASICs shown in Table 1 illustrates why EasyPath is a much superior solution. Unlike structured ASICs, EasyPath customers can get to production volumes much faster and now can do so at lower prices as well.

For more information about the next-generation EasyPath FPGAs, please visit www.xilinx.com/easypath/, where you can get information on the platform support, flexibility features, and use an online cost calculator to find out why EasyPath FPGAs are the lowest total cost solution in the industry. ●●●

Selection Criteria	Structured ASICs*	EasyPath FPGAs
Time to Prototype Samples	4-8 weeks	0 weeks
Total Time to Volume Production	12-15 weeks	8 weeks
Vendor NRE/Mask Costs	\$100K-\$200K	\$75K
Design Costs for Conversion	\$250K-\$300K	\$0
Additional Cost of Tools for Conversion	\$100-\$200K	\$0
Unit Costs	Low	Low
Risk	High	Low
Flexibility to Make Changes In-System	Inflexible	Flexible
Design Conversion from Prototype to Production	Additional Engineering	Conversion Free

*Xilinx market analysis

Table 1 – EasyPath FPGAs versus structured ASICs

inflexible custom solution like standard cell or structured ASICs. Now, with EasyPath FPGAs, Xilinx offers two flexibility features that allow you to enjoy some of the FPGA advantages when you go to volume production at prices below structured ASICs.

Spartan-3 and Virtex-4 EasyPath FPGAs enable you to buy a custom device that supports two applications – one for diagnostic testing and one for the actual application. EasyPath FPGAs can now be tested for two designs, or two variations of the same design. This means that you can

Coming Soon to a location near you!

X-fest

2005

April through June

REGISTER NOW!

**Learn how the latest Xilinx technology can
help you design cost effective solutions faster.**

Gain hands-on experience to speed up your next development cycle.

***What Are You Waiting For?
Register now for the event nearest you.***

Visit www.memec.com/xfest-2005



ASIA  CANADA  EUROPE  JAPAN  UNITED STATES

Copyright 2004 Memec, LLC. All rights reserved. Logos are owned by their proprietors and used by Memec with permission. All company and product names may be trademarks of their respective companies.



The Virtex-4 Power Play

The latest Xilinx FPGA offers revolutionary power innovations.



by Matt Klein
Sr. Staff Engineer, Applications Engineering,
Advanced Products Division
Xilinx, Inc.
matt.klein@xilinx.com

Device power consumption is a primary issue in the semiconductor industry – as process technologies get smaller and faster, they normally consume more power, putting power concerns and performance at odds. The new Virtex-4™ FPGA family from Xilinx® employs innovative architectural features and clever IC design techniques that dramatically reduce power consumption, without compromising performance. This bucks expected trends nor-

mally associated with the reduced feature sizes of 90 nm process technology.

In this article, we'll explore how Xilinx IC designers achieved remarkable power efficiency in the high-performance Virtex-4 FPGA.

Components of Power Consumption

There are two main components to power consumption: static and dynamic. Static or quiescent power is mainly dominated by transistor leakage current. When this current is listed in data sheets, it is listed as I_{CCINTQ} and is the current drawn through the V_{CCINT} supply powering the FPGA core.

Dynamic or active power has components from both the switching power of the core of the FPGA and the I/O being switched. The

dynamic power consumption is determined by the node capacitance, supply voltage, and switching frequency and governed by the basic formula $P=CV^2f$.

Both static and dynamic power have been significantly reduced in Virtex-4 devices, even when compared to Virtex-II Pro™ devices.

Dramatic Power Reduction

The Virtex-4 product family has reduced power consumption in several key areas. The power-per-CLB has been cut in half, with static power reduced by 40% and dynamic power reduced by 50% when compared to the 130 nm Virtex-II Pro FPGA and other 90 nm FPGAs. Furthermore, certain hard-logic silicon functions in the Virtex-4 FPGA reduce power consumption by 80-95%, a whopping factor when compared to the same functions implemented in configurable logic blocks and programmable interconnect routing.

Additionally, comprehensive power planning tools are available to help you get an idea, up front, of power consumption for your Xilinx FPGA under its operating conditions.

Reduced Power Consumption Benefits

Reduced power consumption benefits cut across a few areas of product design in reduced thermal concerns as well as eased power supply design (see Figure 1).

- Reduced thermal concerns – When you reduce power consumption in a device or system, you use smaller heat sinks, or no heat sinks at all in some cases. You also have simpler thermal system design from the point of view of reducing airflows and fan size needs.
- Easier power supply design – You can also use smaller supply circuitry and reduce the number of components in the power supply. Using less PCB space allows you to reduce the cost of the power system. Plus, by not having your device consume as much power, you can achieve higher reliability by lowering the temperature of the FPGA die.

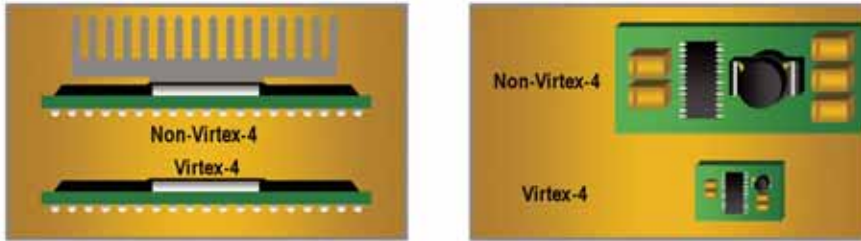


Figure 1 – Virtex-4 devices reduce thermal concerns and simplify power supply design.

Static Power Trends in 90 nm Technology

The reduction in transistor size in 90 nm technology has several effects on power consumption. The biggest potential problem is in the area of static power.

Scaling Trends for Static Power

As we mentioned earlier, static power is dominated by transistor leakage current. Unfortunately, channel leakage increases as transistor size decreases. This is especially true for low V_T transistors where V_T refers to voltage threshold between the gate and drain.

Low V_T transistors are the fastest transistors – the ones with the shortest turn-on and propagation delay – that IC designers use inside the FPGA when the highest speed performance is needed. Regular V_T transistors are also used when less performance is acceptable, but this only helps so much with leakage.

Figure 2 shows that leakage goes up dramatically when moving from 130 nm to 90 nm technology. The Virtex-II Pro device uses 130 nm process technology, whereas the new Virtex-4 device uses 90 nm process technology.

Triple-Oxide – The Savior of Static Power

Triple-oxide simply means that we use a third thickness of oxide in making some of the transistors in the FPGA (two oxide thicknesses are used in devices like the Virtex-II Pro FPGA). Most transistors in the past had a thin oxide layer. Within those transistors could be low V_T , regular V_T , NMOS, or PMOS transistors. Thick-oxide transistors are mostly used for I/O drivers and a few other functions.

Oxide deposition thickness is a very stable and controllable process in the semiconductor industry because it depends on temperature, concentration, and exposure

time. Figures 3a and 3b show the Virtex-4 transistor with the middle oxide thickness used in the triple-oxide process. You may notice that the oxide thickness is still very, very thin, but this thicker oxide transistor has much lower leakage than the standard thin-oxide low V_T and regular V_T transistors used in Virtex-II Pro FPGAs and in various parts of Virtex-4 FPGAs.

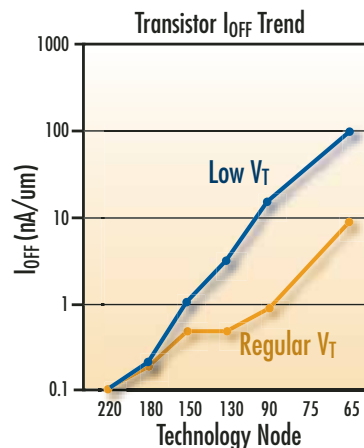


Figure 2 – Transistor leakage trends due to process scaling

Why Doesn't Everyone Use Triple-Oxide?

If triple-oxide is such a great process, why don't other companies like Intel™ or IBM™ use it in their own ASICs?

They probably would if it benefited them. The reason they don't is that all of their transistors need to run at speed; hence, they must use the low V_T leakier transistors for everything. FPGAs can have many different transistor types, which can be selected for function, power, or performance.

FPGAs can use different transistor types for different functions, and Xilinx designers have accomplished this balance.

Optimizing Performance and Leakage

Our IC designers have many things that they can do to adjust the mix to optimize for certain factors. The Virtex-4 FPGA is the first Platform FPGA designed for high speed and low power.

Low V_T transistors are used only where necessary for maximum speed, while the middle thickness of oxide from the triple-oxide process may be used for less aggressive performance with very low leakage. You may use different sizes and types of transistors for performance and function. Combinations are also possible, such as small and medium-sized low V_T fast transistors and small and medium-sized middle oxide thickness transistors. It is not a one-size-fits-all procedure.

Xilinx IC designers were given a directive to reduce power, among other things, in the Virtex-4 platform while maintaining the highest system performance. These transistors are used across the various FPGA functions of LUTs, I/O, interconnect, and configuration memory cells. Even within a given FPGA function, all transistors don't need to be the same, and that is up to the Xilinx IC designers (see Figure 4).

The surprising result of this balancing is that the overall static current in Virtex-4 devices with 90 nm process is reduced by 40% when compared to Virtex-II Pro devices with 130 nm process. Table 1 shows a chart of the weighted average changes to the transistors in the Virtex-4 die compared to Virtex-II Pro die, which allows you to arrive at the reduced transistor leakage in the Virtex-4 FPGA.

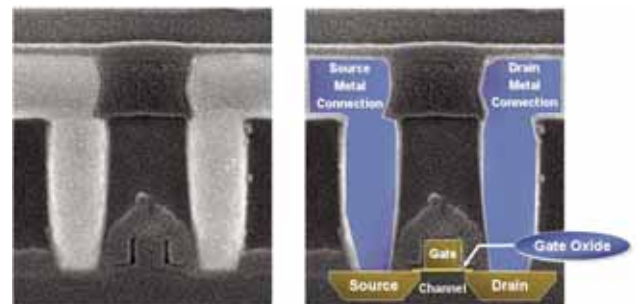


Figure 3a, 3b – Middle oxide thickness Virtex-4 transistor used in triple-oxide process and with highlighted portions of the transistors

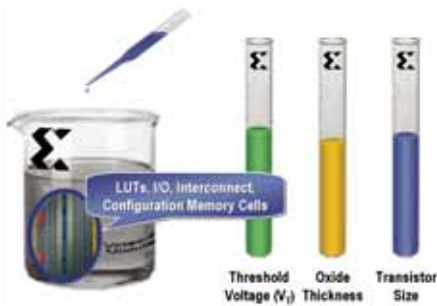


Figure 4 – Optimal transistor mix for minimizing leakage and maximizing performance

Dynamic Power Reduction

Static power reduction, while dramatic, is not the only power winner that you can take advantage of. Dynamic power is also reduced by 50% when compared to Virtex-II Pro FPGAs.

The dynamic power in the FPGA is governed by the following equation:

$$P_{Dynamic} = FPGA_{Core}(CV^2f) + FPGA_{I/O}(CV^2f)$$

The Virtex-4 family of FPGAs has the following:

- Reduced FPGA core dynamic power
- Internal operating voltage is the dominant factor
- Secondary scaling by frequency (f) and node capacitance (C)
- Constant FPGA I/O dynamic power
- Unchanged voltage swing ($V_{I/O}$), toggle rate (f), and pin/pad capacitance (C) for a given I/O standard

So you can see that we may be able to have an effect on dynamic power inside the device, but that dynamic power consumed by I/O switching remains unchanged.

When we go from the 130 nm process of the Virtex-II Pro FPGA to the 90 nm process of the Virtex-4 FPGA, the internal supply voltage changes from 1.5V to 1.2V. This reduces the dynamic power consumption for every internal transistor by 36% ($1 - \left[\frac{1.2}{1.5}\right]^2$) of that in the Virtex-II Pro FPGA.

Additionally, the FPGA internal composite capacitance is reduced in the Virtex-4 FPGA. This internal capacitance comprises transistor parasitic capacitances and trace-to-metal and trace-to-trace capacitances for the interconnecting metal traces. Figure 5 shows the capacitance involved relative to their structures.

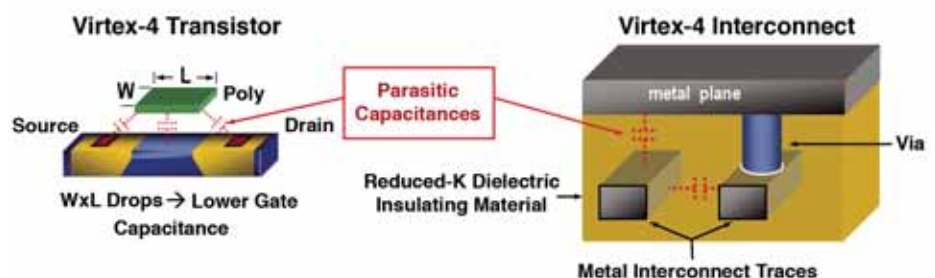


Figure 5 – Internal FPGA capacitance comprises parasitic transistor and interconnect capacitances

Does low-K reduce power? Low-K refers to the dielectric insulating material between the metal traces in the FPGA. Lower K dielectric insulating layers do reduce internal capacitances per unit trace length, but “low-K” is a relative term. Xilinx has reduced-K-insulating materials, and in the past has used low-K itself; we may do so again in the future.

in the transistors; the architecture of the interconnect paths and actual path lengths; and the number of hops through interconnect switches. Xilinx has reduced the overall capacitance for those components in the Virtex-4 FPGA.

The overall effect is mostly due to reduced gate capacitance and lowers capacitance by 20% for Virtex-4 FPGAs when compared to Virtex-II Pro FPGAs. Table 2 shows a dynamic power reduction of 50% for the Virtex-4 FPGA when compared to the Virtex-II Pro FPGA. We have a 23% reduction in dynamic power when running at a 50% higher frequency.

Because the Virtex-4 FPGA is a much higher performance device than the Virtex-II Pro FPGA, you may need to operate it at higher clock speeds to meet newer demanding performance targets that could never be achieved in previous systems.

Parameters	Virtex-II Pro	Virtex-4	Change
Channel Width Ratio	1.00	0.64	-36%
Channel Length Ratio		0.71	-29%
Leakage Current per Unit Width Ratio		1.14	+14%
Leakage Current per Transistor		0.74	-26%
V_{CCINT} Ratio		0.80	-20%
Static Power per Transistor Ration ($I_{LEAKAGE} * V_{CCINT}$)		0.59	-41%

Table 1 – Overall weighted average transistor leakage and parameter comparisons for 90 nm Virtex-4 transistors relative to 130 nm Virtex-II Pro transistors

Parameters	Virtex-II Pro	Virtex-4	Change
V_{CCINT}	1.5	1.2	-20%
C_{TOTAL} (rel.)	1.0	0.8	-20%
f_{MAX} (rel.)	1.0	1.5	+50%
Power at Same Frequency	2.25	1.15	-49%
Power at f_{MAX}	2.25	1.73	-23%

Table 2 – Chart showing changes in internal FPGA in Virtex-4 devices compared to Virtex-II Pro devices and the effect on dynamic power

Parameters	Virtex-II Pro	Virtex-4	Logic Slice Reduction	Logic Slice Power Reduction
QDR II SRAM Interface	550 slices	125 slices	77%	89%
SPI-4.2 Core	5000 slices	3900 slices	22%	61%

$$\text{Logic slice power reduction} = 100 * \left(1 - 0.5 \frac{\text{Virtex-4 slice count}}{\text{Virtex-II Pro slice count}} \right) \%$$

Note: The factor of 0.5 above comes from the fact that Virtex-4 power per slice is 1/2 of the Virtex-II Pro power per slice because of the 50% dynamic power reduction in Virtex-4 devices compared to Virtex-II Pro devices.

Table 3 – QDR II SDRAM and SPI-4.2 core benefit in reduced power consumption from significant logic cell reduction due to new Virtex-4 ChipSync block

Embedded Blocks

Another major area of improvement in power consumption is in the area of embedded functions. This has always been a strength in Xilinx FPGAs, but it is more so in the Virtex-4 FPGA, even when compared to the feature-rich Virtex-II Pro FPGA.

In Virtex-4 FPGAs you can take further advantage of both static and dynamic power reduction by using the embedded functions, which are built as hard-logic functions.

When embedded functions are implemented as hard-logic functions instead of in configurable logic blocks and programmable interconnects, there is a lot less static and dynamic power consumed. This is because far fewer transistors are used for hard, fixed logic than for programmable logic. Additionally, there are no transistors needed to make connections for interconnects in the embedded functions, because there are no programmable interconnects.

Xilinx has carefully studied some of the functions that engineers like you have struggled with and that we have also found tedious to implement within the

FPGA programmable logic. The new embedded functions lower power by 80-95% compared to their configurable logic blocks and routed counterparts in programmable silicon.

Comprehensive Power Planning Tools

Another useful thing in planning power is that Xilinx data sheets show you both typical and maximum power consumption numbers. Maximum numbers are for worst-case process, temperature, and voltage, but many designers are very happy to work with typical numbers, depending on their application and the number of parts being used in one system.

One additional very useful thing that you can take advantage of in planning for power consumption in Xilinx FPGAs are power planning tools. Xilinx web power tools are available for estimating power early in the design cycle. Also, as part of the Xilinx design flow, XPower looks in more detail at a mapped or routed design. Both can be found, along with power application notes, by searching the Xilinx website for the phrase “Xilinx Power Tools.”

Conclusion

Xilinx has made profound improvements in both static and dynamic power in the Virtex-4 90 nm family of FPGAs when compared to Virtex-II Pro FPGAs – and (we believe) in comparison to our competitors. We have done this through a multi-pronged, purposeful approach in the areas of reduced leakage current, reduced dynamic power consumption, and embedded functions, without compromising performance. These, along with comprehensive power planning tools, make the Virtex-4 device an excellent choice for a high-performance FPGA system.

For more information about power consumption in Virtex-4 and other Xilinx FPGAs, visit www.xilinx.com/products/design_resources/design_tool/grouping/power_tools.htm. 

Virtex-4 Embedded Functions and Reduction of Dynamic Power

- PowerPC – 50% power reduction compared to Virtex-II Pro PowerPC
 - 10:1 power reduction over FPGA fabric-built version
- DSP – XtremeDSP™ slice greatly reduces logic cells, which previously needed many filtering functions
 - 20:1 power reduction over Virtex-II Pro separated multiply/accumulate functions
- SSIO – New ChipSync™ block reduces logic cell count for SSIO (source synchronous I/O) designs
 - Significant logic cell savings for various memory and networking interface designs leads to reduction in overall power up to 9:1 for selected designs (see Table 3)
- Embedded Ethernet MAC(s) – No need to use logic and interconnect for MAC function, which saves >3,000 logic cells for the Xilinx implementation
- FIFO – SmartRAM™ memory includes built-in FIFO controllers, which can save hundreds of logic cells per FIFO and greatly simplify design as well



Deliver Efficient SPI-4.2 Solutions with Virtex-4 FPGAs

Virtex-4 devices offer an ideal platform for source-synchronous designs like the widely adopted SPI-4.2 interface.

by Chris Ebeling
Principal Engineer
Xilinx, Inc.
chris.ebeling@xilinx.com

Krista Marks
Sr. Manager, IP Solutions Division
Xilinx, Inc.
krista.marks@xilinx.com

SPI-4.2 (System Packet Interface Level 4 Phase 2) is the Optical Internetworking Forum's recommended interface for the interconnection of devices for aggregate bandwidths of OC-192 (ATM and POS) and 10 Gbps (Ethernet), as illustrated in Figure 1.

In the last few years, this interface has become the de-facto standard on all leading 10 Gbps framer ASSPs and has been implemented directly on many next-generation network processors. SPI-4.2 has been broadly adopted because of its efficient interface, which offers high bandwidth with a low pin count and seamless handling of typical system requirements such as flow control, error insertion/detection, synchronization, and bus re-alignment.

The Xilinx® Virtex-4™ architecture provides an ideal platform for implementing SPI-4.2. The Xilinx SPI-4.2 LogiCORE™ IP targeting Virtex-4 devices provides a solution with one-third less resources, dramatic power savings, 1+ Gbps LVDS double-data-rate (DDR) I/O, and complete pin assignment flexibility.

SPI-4.2 LogiCORE IP

Xilinx has improved on its Virtex-II™ and Virtex-II Pro™ SPI-4.2 solution, already one of the smallest in the industry, and made it 30% smaller by leveraging new ChipSync™ technology in the Virtex-4 FPGA. ChipSync technology is supported on every pin of the Virtex-4 device family; thus the new SPI-4.2 LogiCORE IP can be targeted to any device pin-out. This allows you to select I/O pins that best fit your system and PCB requirements.

In addition, for those applications requiring multiple SPI-4.2 interfaces, the Virtex-4 FPGA's logic density, high pin count, and extensive clocking resources will support four or more full-duplex cores in a single device. Regardless of the performance your application requires,

Virtex-4 devices fully support the entire SPI-4.2 operating range, with high-speed LVDS support of data rates greater than 1 Gbps per pin.

ChipSync Technology

Xilinx introduced ChipSync technology in Virtex-4 FPGAs to enhance I/O capability when used for source-synchronous applications like SPI-4.2. ChipSync features are supported in every Virtex-4 I/O pin and include:

- New serial and de-serial (OSERDES and ISERDES) features. This enables logic built in the fabric to interface to the I/O at a fraction of the source-synchronous clock rate. The ISERDES also includes a Bitflip function. Bitflip allows you to shift the starting bit of deserialized data to achieve proper word alignment when linking multiple pins together (bus deskew).
- A new input delay (IDELAY) feature. This allows you to precisely adjust the input delay of each bit of a bus independently, in 78 ps increments. This provides a mechanism for tuning the interface timing to the system environment.

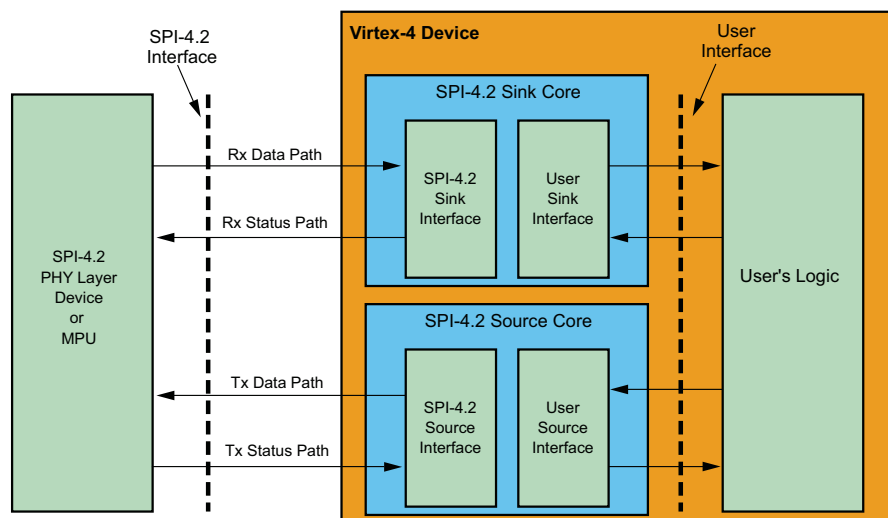


Figure 1 – Typical SPI-4.2 application

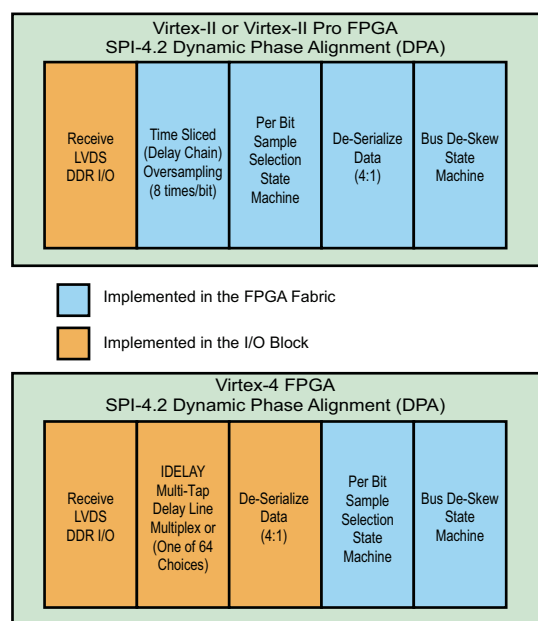


Figure 2 – DPA implementation in I/O logic for Virtex-II devices versus Virtex-4 devices

Additional DDR registers are now fully integrated into the input (ILOGIC) and output (OLOGIC) pins, simplifying the interface between the FPGA fabric and I/O blocks and supporting data transfer to and from the I/O logic on a single clock edge.

SPI-4.2 and ChipSync Technology

The SPI-4.2 interface has a DDR source-synchronous data bus that comprises 18 LVDS pairs (16 data bits, 1 control bit, and 1 clock). The SPI-4.2 source-synchronous clock varies from 311 MHz to 500 MHz.

For example, a typical OC-192 framer will require an aggregate bandwidth of 10 Gbps, which for a 16-bit dual data rate bus would require a data clock of at least 311 MHz, with 350 MHz a typical clock rate. The Xilinx SPI-4.2 LogiCORE IP easily meets your application requirements, regardless of performance, and with Virtex-4 ChipSync technology delivers a solution that is smaller and more flexible than prior FPGA implementations.

The SPI-4.2 core uses ChipSync technology to serialize egress data and de-serialize ingress data to a four-word (bus cycle) SPI-4.2 data stream at a lower clock rate. Operation of the core logic at a lower internal clock rate

allows you to implement high-frequency SPI-4.2 interfaces in the slowest speed grade Virtex-4 device.

The ISERDES and OSERDES functions allow the core logic to time multiplex and de-multiplex these four words to and from the I/O logic without using any CLB logic resources. The core logic need only operate at half the source-synchronous DDR clock rate. For example, a SPI-4.2 interface with a 500 MHz DDR reference clock would only require an FPGA fabric clock of 250 MHz – easily achievable in the Virtex-4 architecture.

As the frequency of the source-synchronous clock increases, data recovery at the receiving (sink) device becomes more challenging. The SPI-4.2 protocol provides a calibration data, or training pattern, that permits a receiving device to adjust its data sampling to the system interface timing. The process of tuning the interface to its particular timing is referred to as dynamic phase alignment (DPA).

Before Virtex-4 devices, Xilinx DPA solutions worked by over-sampling the input data and choosing the best sample from the group. This required valuable FPGA resources and careful control of the input data path in the FPGA fabric, restricting the SPI-4.2 interface pin placement. In Virtex-4 FPGAs, the IDELAY feature present in every I/O is ideally suited to perform this function, as shown in Figure 2. (See “Dynamic Phase Alignment with ChipSync Technology in Virtex-4 FPGAs,” also in this issue of the *Xcell Journal*).

The IDELAY features have two primary benefits for the SPI-4.2 core in Virtex-4 FPGAs:

- Integrating the IDELAY feature into the input pin (ILOGIC) reduces the FPGA resources required for DPA to less than 350 slices.
- The IDELAY function's ability to adjust the data sampling point enables DPA to be implemented in the I/O – except for a small control state machine, which is implemented in the fabric. The state machine portion is fully synchronous and does not require a complex macro. Thus, there are no restrictions on SPI-4.2 pin assignments.

Clocking Resources

Virtex-4 FPGAs provide an unprecedented number of clock resources for implementing multiple SPI-4.2 interfaces in a single device. With the Virtex-II and Virtex-II Pro architectures, implementing more than two SPI-4.2 interfaces posed a clock management challenge. The abundance and flexibility of clock distribution in the Virtex-4 family solves this challenge, supporting as many SPI-4.2 interfaces as the device logic and I/O will allow.

In Virtex-4 devices, all devices have 32 global clock resources. No restrictions exist on global clock distribution other than a maximum of eight global clocks per clock region. All clock regions have access to any 8 of the 32 total global buffers, regardless of the requirements of other clock regions.

In addition to the eight global clocks, each region in the device has two regional clock buffers. The regional clock resources are ideal for interface clocking, like the source-synchronous clock scheme used by SPI-4.2. Note that even the smallest Virtex-4 device has a total of 48 available clock resources, each designed for low-skew clock distribution and clock power management. The SPI-4.2 LogiCORE IP can be configured to use either global or regional clock resources.

In Virtex-4 FPGAs, the global clock trees and associated buffers are implemented differentially, for best duty-cycle fidelity and greater common-mode noise rejection. With Virtex-II and Virtex-II Pro devices, if SPI-4.2 interface operates above 350 MHz, you must route the high-speed reference clock using two clock buffers to minimize duty-cycle distortion at the DDR registers.



Figure 3 – Illustration of four SPI-4.2 LogiCORE IP implemented on a Virtex-4 XC4VLX60 device

interfaces in the larger devices (Figure 3). The Virtex-4 clocking capability opens up a whole new class of SPI-4.2 applications, and provides an ideal platform for applications such as multiplexing and de-multiplexing, bridges, and switches.

	VIRTEX-II	VIRTEX-II PRO	VIRTEX-4
Power: Static Alignment @ 700 Mbps per LVDS Pair	1.9W	1.75W	1.55W
Power: Dynamic Alignment Performance per LVDS Pair	2.6W @800 Mbps	2.8W @944 Mbps	2.0W @1 Gbps
Speed Grades Supporting 800 Mbps per LVDS Pair	-6	-6, -7	-10, -11, -12

Table 1 – SPI-4.2 power estimates for Virtex-II, Virtex-II Pro, and Virtex-4 FPGAs

Because each global clock tree in Virtex-4 FPGAs is implemented differentially, only one clock buffer is required.

Not only does the Virtex-4 architecture have considerably more clock resources, but because they are distributed differentially, the SPI-4.2 LogiCORE IP requires fewer of them. These high-performance clock resources support as many as four SPI-4.2 interfaces in a mid-range device (LX40/LX60) and more than four SPI-4.2

the enhanced 90 nm semiconductor process and because the LogiCORE IP uses 30% less fabric resources. At the same time, Virtex-4 FPGAs support 30% higher internal performance for SPI-4.2, with a maximum frequency of 250 MHz in the lowest speed grade (compared to 175 MHz in the lowest speed grade of Virtex-II and Virtex-II Pro devices). In addition, Virtex-4 FPGAs support 1+ Gbps LVDS for every I/O on the device.

This means that not only can you place multiple SPI-4.2 interfaces anywhere on the device, but for each implemented interface you get an aggregate bandwidth as high as 16+ Gbps. Designs that do not require this level of performance (such as more typical framer interfaces running at 10-12 Gbps) automatically get additional performance overhead that ensures ease of design integration and timing closure.

Conclusion

The Xilinx SPI-4.2 LogiCORE IP, coupled with Virtex-4 features, provides a highly efficient SPI-4.2 solution. We developed ChipSync technology that supports every I/O pin specifically for source-synchronous interfaces like SPI-4.2.

This technology enables you to design the most efficient SPI-4.2 solution, which uses significantly less resources (35% less), allows fully flexible device pin assignments (you choose the pinout), and supports extremely high interface speeds (1+ Gbps LVDS DDR I/O).

The higher performance is even more compelling because Virtex-4 FPGAs deliver it with lower power and significantly higher internal operating rates. The wealth of Virtex-4 clocking resources, combined with full pin assignment flexibility, opens up the possibility for new applications with multiple SPI-4.2 interfaces.

For more information about SPI-4.2 LogiCORE IP targeting Virtex-4 devices, please refer to this site at the Xilinx IP Center: www.xilinx.com/xlnx/xebiz/designResources/ip_product_details.jsp?key=DO-DI-POS4MC. A hardware demonstration is also available; for more information, contact your Xilinx representative. 

What will happen when your
FPGA and PCB designs meet?

Integrated Systems



Integrated Systems | The potential for disaster is enormous if your designers work independently of one another. Mentor Graphics eliminates that potential with the only integrated systems design solution in EDA. We have superior design solutions in both PCB and FPGA, so it stands to reason that we'd create the only truly integrated system flow. Our unique class of tools empowers designers to concurrently design FPGA's and PCB's. Increase your system performance and design team productivity. Get the systems integration white paper at www.mentor.com/techpapers or call 800.547.3000.

**Mentor
Graphics®**

Virtex-4 Memory Interfaces

Virtex-4 devices make challenging memory interface requirements simple.

by Maria George
Senior Product Applications Engineer
Xilinx, Inc.
maria.george@xilinx.com

Xilinx® Virtex-4™ devices have a 64-tap absolute delay element built in each I/O, making high-speed memory interface read data capture very easy. This feature also provides the flexibility to adopt different read data capture schemes where clock/strobe or data can be delayed.

During a write to the external memory device, the clock/strobe must be transmitted center-aligned with respect to data. A memory write is easy to implement with Virtex-4 devices by means of the quadrature phase outputs of the DCM (CLK0, CLK90, CLK180, CLK270), ensuring that the clock/strobe is center-aligned with data. Figure 1 illustrates the clock/strobe and data phase relationship during read and write transactions.

For most memory interfaces, such as DDR 2 SDRAM, RLDRAM II, FCRAM II, and QDR II SRAM, the data rate is twice the clock rate because data is received and transmitted on both the ris-

ing and falling edges of the forwarded clock/strobe. Virtex-4 devices have both input and output DDR flip-flops, making DDR operation extremely simple.

Write Data and Clock/Strobe Transmission

During a write operation, the clock/strobe is generated using the output DDR registers clocked by a DCM clock output (CLK0) on the global clock network. The write data is transmitted using the output DDR registers

clocked by a DCM clock output that is 90 degrees phase ahead (CLK270) of the clock used to generate clock/strobe. This meets the memory vendor specification of centering the clock/strobe in the data window.

Another innovative feature of the output DDR registers is the **SAME_EDGE** mode of operation. In this mode, a third register clocked by a rising edge is placed on the input of the falling edge register (Figure 2). Using this mode, both rising edge and falling edge data can be presented to the output DDR registers on the same clock edge (CLK270), thereby allowing higher DDR performance with minimal register-to-register delay.

Read Data Capture

Most memory interfaces are source-synchronous interfaces, where the clock/strobe is received edge-aligned with data during a read from the external memory device. This makes read data capture challenging because the read clock/strobe must be delayed to capture read data.

Read data capture is challenging because the read data and the incoming memory read clock/strobe are received edge-aligned from the external device.

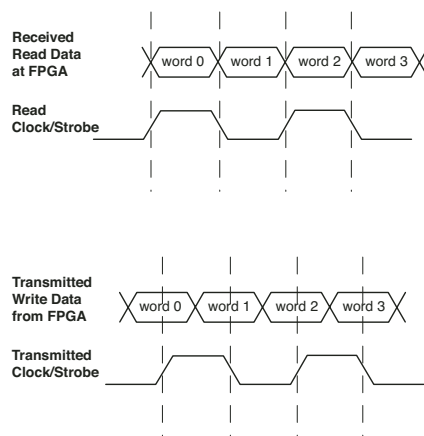


Figure 1 – Clock/strobe and data during read and write

The traditional technique to capture read data is to register it in the delayed memory clock/strobe domain. This entails:

- Ensuring that the memory clock/strobe and the associated data have matched PCB trace delays between the memory device and the FPGA
- Delaying the clock/strobe signals such that the edges of the clock/strobe center in the valid data window, as shown in Figure 3
- Registering the read data with the delayed memory clock/strobe
- Synchronizing registered read data to the system (FPGA) clock domain

An alternate and simpler technique, currently used in Xilinx reference designs, is to capture read data directly in the system (FPGA) clock domain. This entails:

- Ensuring that the memory clock/strobe and the associated data have matched PCB trace delays between the memory device and the FPGA
- Determining phase difference between the memory clock/strobe to the system (FPGA) clock by detecting two memory clock/strobe transitions in the system clock domain
- Detecting transitions of memory clock/strobe after the memory initialization sequence by delaying memory clock/strobe with respect to the system (FPGA) clock in unit increments
- Delaying read data based on memory clock/strobe to system (FPGA) phase information such that the system (FPGA) clock is centered in the valid data window

Both techniques require delay elements to delay the clock/strobe or data.

The 64-tap, 80 ps absolute delay element available in each Virtex-4 I/O allows center alignment of memory clock/strobe in the data window or data centering with the system (FPGA) clock. Each Virtex-4 I/O also has input DDR flip-flops that are required for read data capture, either in the delayed memory

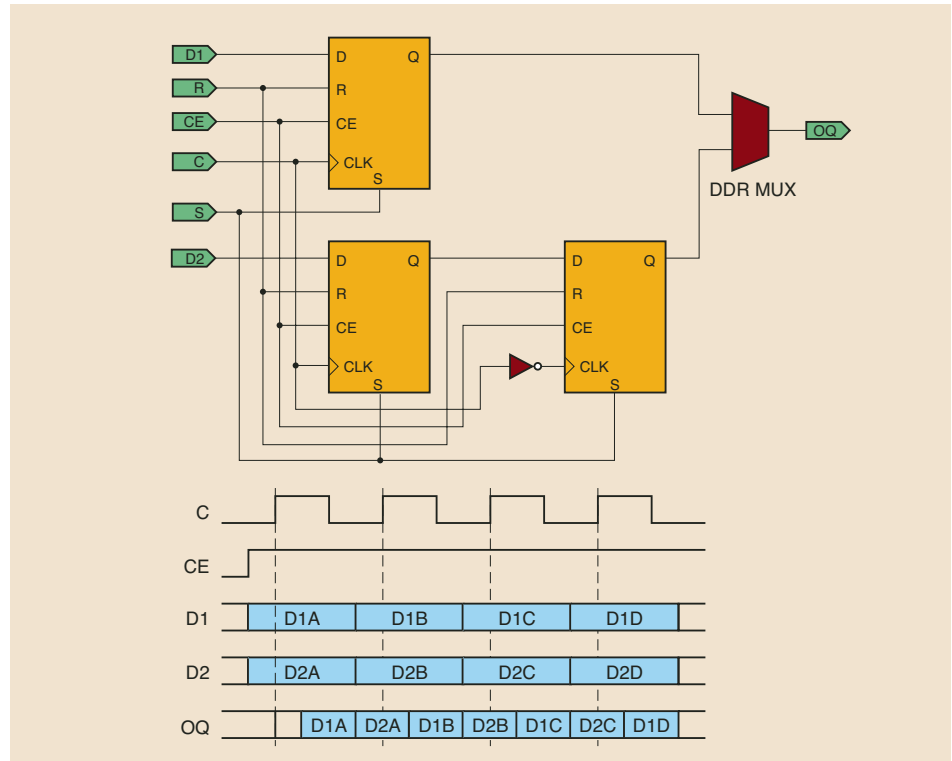


Figure 2 – Output DDR in SAME_EDGE mode

strobe domain or the system (FPGA) clock domain.

You can use the input DDR flip-flops in the **SAME_EDGE** or **SAME_EDGE_PIPELINED** modes. In the **SAME_EDGE** mode, the falling edge data is output on the following rising edge of the clock (Figure 4). In the **SAME_EDGE_PIPELINED** mode, both the rising edge and falling edge data are output together on the same rising edge of the clock (Figure 5). With these modes you can achieve higher design performance by avoiding half-clock cycle data paths in the FPGA fabric.

In the first technique, read data is captured in the delayed memory clock/strobe

domain and must be re-captured in the system (FPGA) clock domain. The transfer of captured read data from the delayed memory clock/strobe domain to the internal system (FPGA clock) domain is defined as read data re-capture. Read data is re-captured within the I/O block.

Using the second technique, implemented in the Xilinx reference designs, you can directly capture read data in the system (FPGA) clock domain by delaying read data to meet the setup/hold time of the flip-flops in the system (FPGA) clock domain. A simple state machine is sufficient to implement the center alignment of the delayed read data with respect to

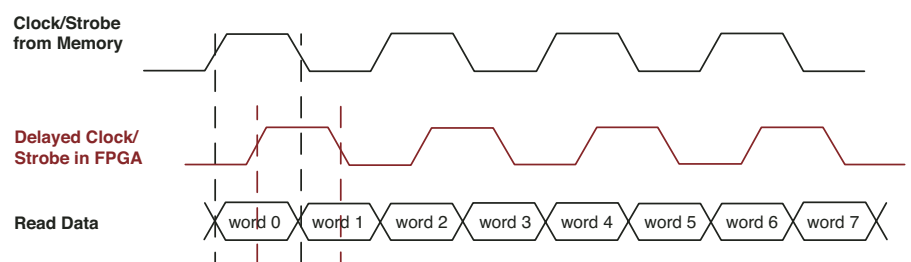


Figure 3 – Clock/strobe delayed in FPGA to center in read data window

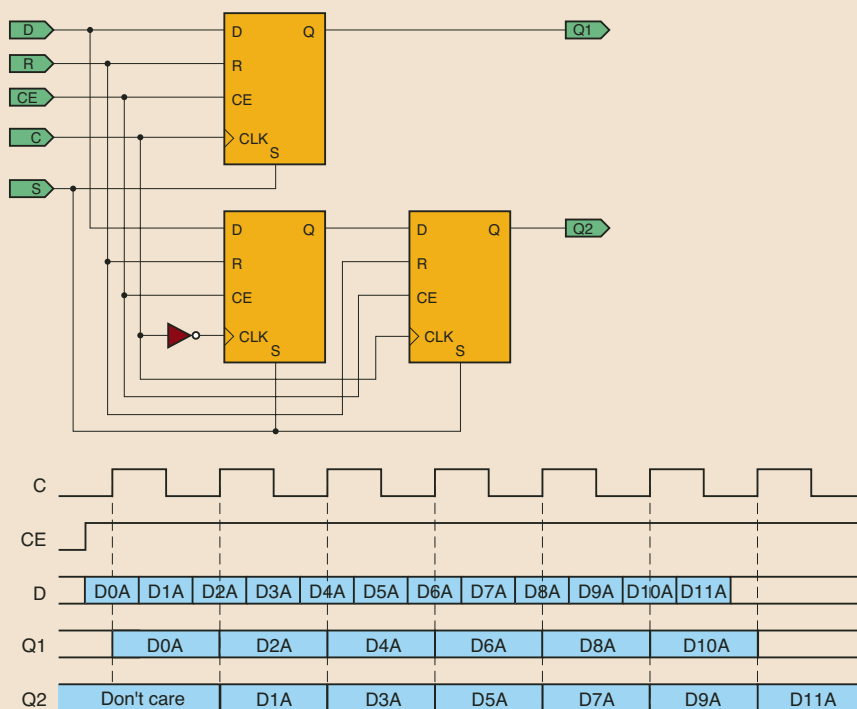


Figure 4 – Input DDR in the SAME_EDGE mode

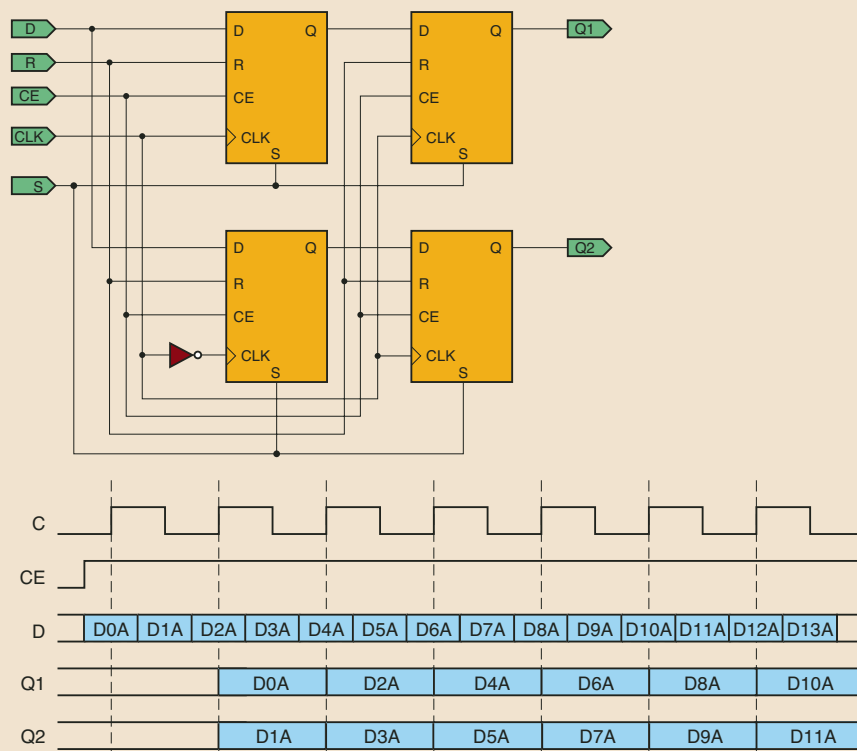


Figure 5 – Input DDR in SAME_EDGE_PIPELINED mode

the system (FPGA) clock after the initialization period.

This “run time” adjustment after the memory initialization sequence has significant advantages over other methods that set the required delay or phase shift during “compile time.” The 64-tap absolute delay element compensates for variations in process, temperature, or voltage, and hence increases the timing margins – resulting in a more reliable system.

The read data is re-captured and stored directly into the block RAM FIFO, a Virtex-4 feature that saves additional logic resources.

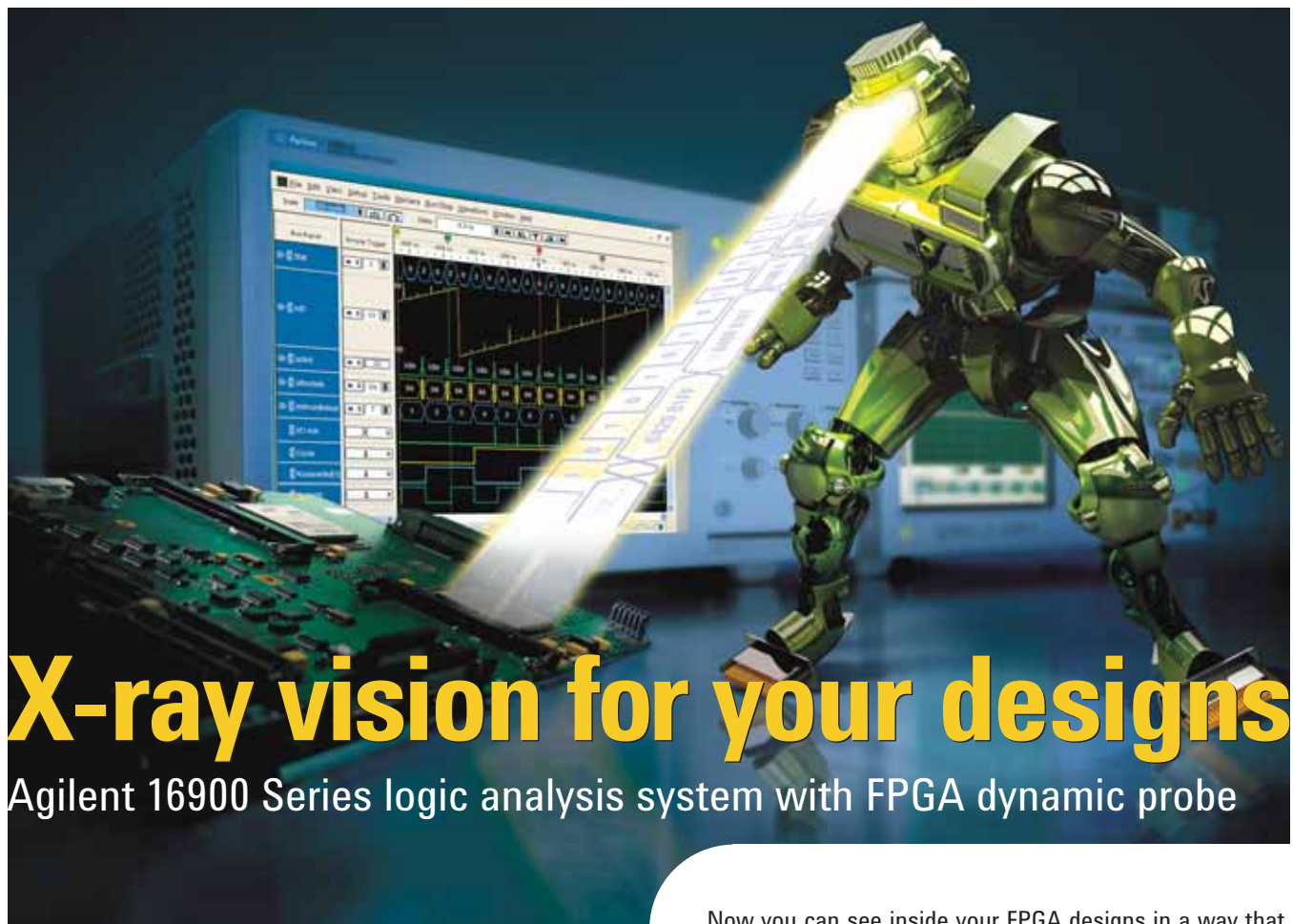
Conclusion

Virtex-4 architectural features enable you to easily and reliably implement high-speed memory interfaces. You can use the 64-tap, 80 ps absolute delay elements to capture read data by either delaying the memory clock/strobe or the data. Built in each I/O, the 64-tap absolute delay elements provide you the flexibility to select any I/O for memory interfaces. The “run time” adjustment after memory initialization improves design margins.

The input and output DDR registers enable you to receive and transmit clock/strobe and data at high frequencies; the differential clocking resource provides higher performance with better duty cycle and lower global clock buffer utilization; and the block RAM FIFO feature enables you to store transmitted or received data without additional logic resources.

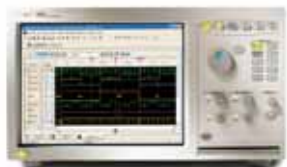
For more information about the implementation and design details of different memory interfaces in Virtex-4 devices, visit the following websites:

- DDR2 SDRAM (XAPP 701 and XAPP702) and DDR SDRAM (XAPP709): www.xilinx.com/products/design_resources/mem_corner/resource/xaw_dram_ddr.htm
- RLDRAM (XAPP710): www.xilinx.com/products/design_resources/mem_corner/resource/rlDRAM.htm
- QDR II SRAM (XAPP703): www.xilinx.com/products/design_resources/mem_corner/resource/xaw_sram_qdr.htm



X-ray vision for your designs

Agilent 16900 Series logic analysis system with FPGA dynamic probe



- Increased visibility with FPGA dynamic probe
- Intuitive Windows® XP Pro user interface
- Accurate and reliable probing with soft touch connectorless probes
- 16900 Series logic analysis system prices starting at \$21,000



Agilent Direct

Get a quick quote and/or FREE CD-ROM with video demos showing how you can reduce your development time.

U.S. 1-800-829-4444, Ad# 7909

Canada 1-877-894-4414, Ad# 7910

www.agilent.com/find/new16900

www.agilent.com/find/new16903quickquote

Now you can see inside your FPGA designs in a way that will save days of development time.

The FPGA dynamic probe, when combined with an Agilent 16900 Series logic analysis system, allows you to access different groups of signals to debug inside your FPGA—without requiring design changes. You'll increase visibility into internal FPGA activity by gaining access up to 64 internal signals with each debug pin.

You'll also be able to speed up system analysis with the 16900's hosted power mode—which enables you and your team to remotely access and operate the 16900 over the network from your fastest PCs.

The intuitive user interface makes the 16900 easy to get up and running. The touch-screen or mouse makes it simple to use, with prices to fit your budget. Optional soft touch connectorless probing solutions provide unprecedented reliability, convenience and the smallest probing footprint available. Contact Agilent Direct today to learn more.



Agilent Technologies

dreams made real



Designing with the Virtex-4 XtremeDSP Slice

Harness the full capabilities of the XtremeDSP slice in filter design.

by Niall Battson
DSP Applications Engineer
Xilinx, Inc.
niall.battson@xilinx.com

With the introduction of Xilinx® Virtex-4™ FPGAs in September 2004, the world of DSP design witnessed a dramatic leap in programmable logic DSP: higher performance, lower cost, lower power, and maximum flexibility.

At the same time this phenomenon asks DSP hardware engineers to change their traditional way of designing and embrace a different approach. These great improvements have been made possible by the XtremeDSP™ slice.

The XtremeDSP Slice

The XtremeDSP slice (also referred to as the DSP48) is a high-performance multiplier and arithmetic unit with great flexibility that can form the building block of many DSP algorithms implemented in FPGAs. A detailed diagram of the DSP48 structure is shown in Figure 1.

The XtremeDSP slice comprises four main sections:

- I/O registers
- 18 x 18 signed multiplier
- Three-input adder/subtractor
- Op-mode multiplexers

The I/O registers ensure a maximum clock performance of 500 MHz in the fastest speed grade device (400 MHz in the slowest speed grade), also ensuring support for higher sample rates. The dynamic op-mode multiplexers are key to the functionality of the structure; they are responsible for the DSP48's great flexibility. For example, in a simple MACC engine, you set the X and Y MUX to multiply and select the feedback path from the registered output P as the Z MUX input to the arithmetic unit.

In the Virtex-4 architecture, XtremeDSP slices are arranged in columns. The most important aspect about the column is the cascade logic and routing between each block, which exists on both the input and output stages of each slice. This dedicated routing enables a number of filters and other functions to be built entirely within the XtremeDSP slice, thus removing the need for signals to be routed through the FPGA interconnect or logic fabric.

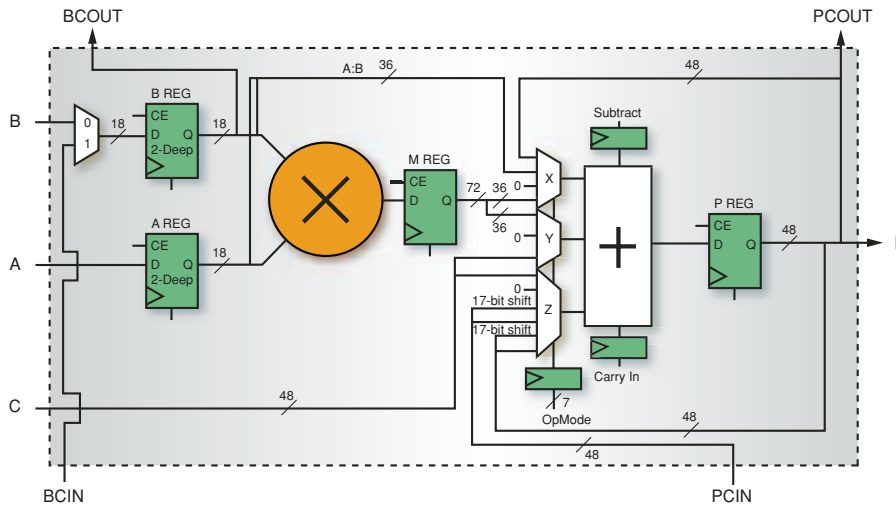


Figure 1 – Simplified diagram of the XtremeDSP slice

However, you must take this adder-chain configuration into account when designing functions that exploit the XtremeDSP slice. Herein lies the fundamental change in the approach to filter design. The simple, traditional adder-tree approach limited the performance and extensibility of a given filter implementation. By using adder-chain-style implementations, these limitations are lifted and the huge benefits Virtex-4 FPGAs offer are possible.

The embedded nature of the XtremeDSP slice has also had a radical impact on reducing the power consumed by high-speed multiply and add functions. Figure 2 illustrates this dramatic reduction, showing that the dynamic power consumption is 1/17 of Virtex-II Pro™ devices with a specification of 2.9 mW/100 MHz. As a designer, you should migrate as much functionality into these embedded functions as possible.

Filter Techniques

During the last ten years, hardware and FPGA designers have created a wide variety of filter architectures to efficiently exploit the building blocks that the current generation of technology offers. With the introduction of Virtex-4 FPGAs and the XtremeDSP slice, filter implementations must change to most efficiently exploit this latest FPGA offering. Filters are prolific in DSP designs and nearly always form the starting point for analyzing an architecture.

The general FIR filter equation is a summation of products (also known as an inner product) defined in the equation:

$$y_n = \sum_{i=0}^{N-1} x_{n-i} h_i$$

In this equation, a set of N coefficients is multiplied by N respective data samples, and the results are summed to form an individual result. The values of the coefficients determine the characteristics of the filter: low-pass, band-pass, or high-pass.

The Semi-Parallel FIR Filter

Even within the filter world, you can implement a wide variety of filters. The key parameters that tell us which FIR filter implementation we will construct are:

- Number of coefficients (N)
- Sample rate (Fs)

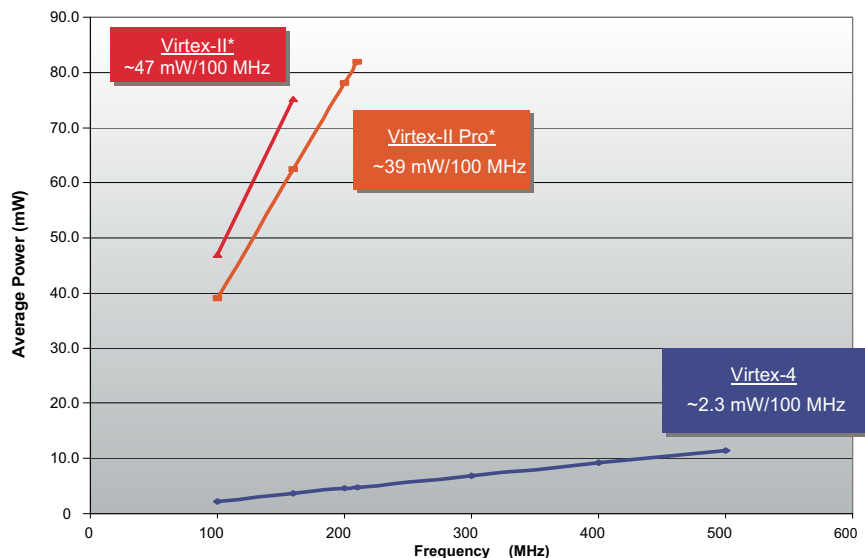
Let's examine a particular filter structure to demonstrate the key design techniques that can help you maximize the benefits of Virtex-4 devices. Our filter has 20 coefficients and a sample rate of 74.25 MHz.

As noted earlier, the maximum capable clock speed of the XtremeDSP slice is 400 MHz in the slowest speed grade (-10). Therefore, we have a total of five clock cycles to perform the required 20 multiply and adds to form the result.

This equation determines how many multipliers to use for a particular semi-parallel architecture:

$$\text{Number of Multipliers} = \frac{\text{Maximum Input Sample Rate} \times \text{Number of Coefficients}}{\text{Clock Speed}}$$

For our example, the required number of multipliers will be four. Once we have determined the required number of multipliers, there is an extendable architecture using the XtremeDSP slices that can serve as the basis for the filter.



Conditions: TT, 25C, nominal voltage, fully pipelined multiply-add mode, random vectors

* Based on power estimator spreadsheet, uses slice logic

Figure 2 – Dynamic power consumption of the XtremeDSP slice

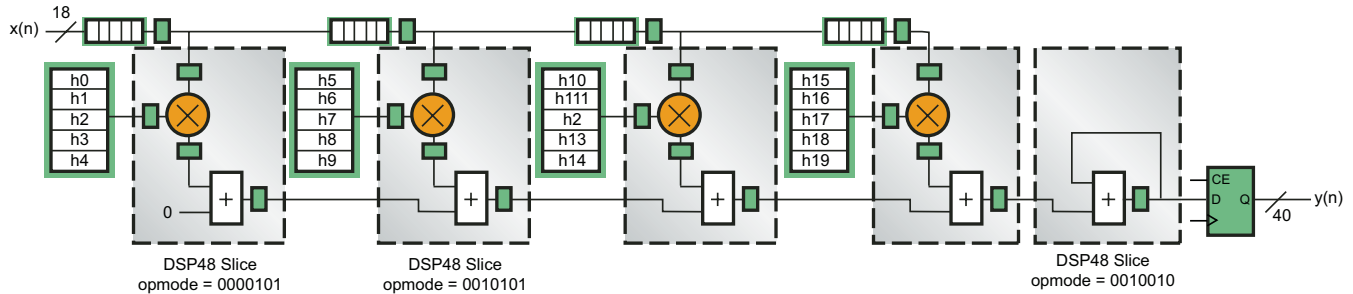


Figure 3 – The four-multiplier semi-parallel systolic FIR filter

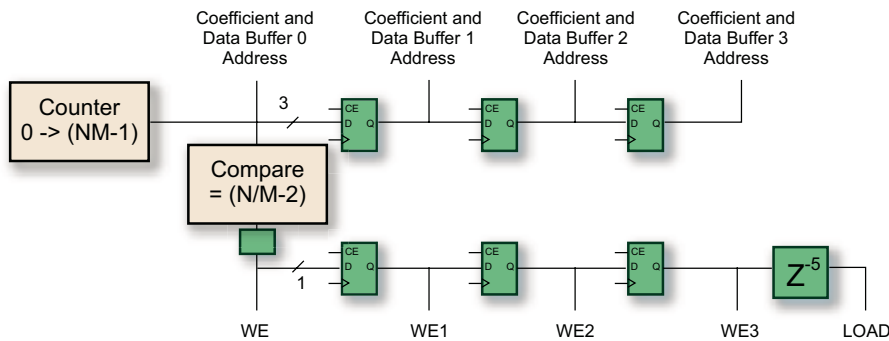


Figure 4 – Control logic for the four-multiplier semi-parallel FIR filter

XtremeDSP arithmetic units are designed to be chained together easily and efficiently thanks to dedicated routing between slices. Figure 3 illustrates how the four XtremeDSP multiply and add elements are cascaded together to form the main part of the filter.

It is critical to highlight the usage of the adder chain here rather than the more traditional adder tree. The adder chain has a profound impact on the control logic required for the filter, as well as its efficiency, because of the mapping to the XtremeDSP slice.

Continuing to analyze the filter structure, an extra XtremeDSP slice is required to perform the accumulation of the partial results, thus creating the final result. A new result is created every five clock cycles. This means that for every five cycles the accumulation must be reset to the first inner product of the next result. This reset (or load) is achieved by changing the op-mode value of the XtremeDSP slice for a single cycle, from 0010010 to 0010000 (this is just a single bit change). At the same time, the capture register is enabled and the final result stored on the output.

The Control Logic

The control is the most important and complicated aspect of semi-parallel FIR filters; getting it right is crucial to filter operation. Because the XtremeDSP slice is most efficiently used in adder chains, memory addressing is necessary to provide the delay for each multiply-add element that the adder chain causes. Figure 4 illustrates the control logic required to create memory addressing.

The counter creates the fundamental zero through four count. This is then delayed by one cycle by the use of a register in the control path. Each successive delay is used to address both the coefficient memory and the data buffer – and their respective multiply-add elements. Hence, a single delay is required for the second multiply-add element, two delays for the third multiply-add element,

and so on. Note that this is extensible control logic for M number of multipliers.

Figure 4 also shows write enable sequencing. A relational operator is required to determine when the count limited counter resets its count. This signal is high for one clock cycle every five cycles, reflecting the input and output data rates. The clock enable signal is delayed by a single register just like the coefficient address; each delayed version of the signal is tied to the respective section of the filter.

The filter and control logic are extremely cascadable. The address for each SRL16E data buffer and coefficient memory pair are a delayed version of the previous elements' address, and are identical.

The performance and resource utilization for our filter is specified in Table 1. In the table, you can see how logic slice utilization dramatically drops when using the XtremeDSP slice. Clock frequency performance approximately doubles over Virtex-II Pro FPGAs.

Four-Multiplier 20-Tap Semi-Parallel FIR Filter 18-Bit Data, 18-Bit Coefficients	Virtex-4 (-11)	Virtex-II Pro (-7)
Logic Slices	108	309
XtremeDSP Slice	5	
Embedded Multipliers		7
Performance (Sample Rate)	90 MHz	77 MHz
Performance (Clock Frequency)	450 MHz	231 MHz

Table 1 – Resource utilization and performance of four-multiplier 20-tap semi-parallel FIR filter

Three Important Design Points

This new filter architecture, along with Virtex-4 devices and the XtremeDSP slice, addresses the demanding needs of current and future DSP designs. However, it is only one filter in an extremely large array of possible implementations, not to mention other DSP functions such as IIRs, FFTs, and DCTs.

Knowing this, you can take away three very important design questions that will enable you to exploit the XtremeDSP slice and Virtex-4 device as designed.

1. Is the design running as fast as possible?

The fastest speed grade (-12) should run at 500 MHz. If your design is running at 50 MHz, you've got the room to reduce your resource utilization by increasing performance (and reducing cost) by making more efficient use of the FPGA resources. The faster a particular function operates, the smaller it becomes. Our semi-parallel FIR filter, for example, used five XtremeDSP slices running at 375 MHz instead of 20 XtremeDSP slices running at 74.25 MHz.

2. Are there any XtremeDSP slices left?

If you are not using them all up, you can probably add some functionality. This can lead to logic slice reduction and lower power consumption.

3. Are you using adder chains instead of adder trees?

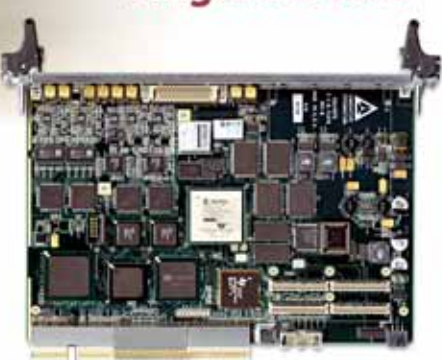
DSP algorithms must aim to exploit adder chain-based implementations wherever possible, as this will lead to the best utilization of the XtremeDSP slice. Such implementations will result in performance gains, power reduction, and logic slice reduction.

Conclusion

For more information, see the XtremeDSP Slice Design Considerations User Guide, which provides in-depth details on other filter implementations and DSP functions, at www.xilinx.com/bvdocs/userguides/ug073.pdf. There are also other HDL and System Generator for DSP reference designs to get you started. 

Onward to Glory

Complete Software Radio Solution on a single 6U card

Features

- ▶ 1GHz TMS320C6416 DSP
- ▶ 105 MHz, 14 bit 2 Ch. Analog I/O
- ▶ 32 MB RAM
- ▶ 6 Million gate Virtex II FPGA
- ▶ 32/64-bit cPCI bus
- ▶ PMC expansion site
- ▶ STAR Fabric interface

Quixote

Get your data sheets now!
www.innovative-dsp.com/quixote
 805.520.3300 phone • 805.579.1730 fax


Innovative Integration
... real time solutions!

Four Screamers

...four screameing fast C6416 DSPs, that is



Radio frequency signal processing solution



▶ Four, 720 MHz TMS320C6416 DSPs each with:

- 64MB SDRAM
- Dedicated 200 MB/s links between DSPs and FPGAs

▶ Two, 4M Gate User-Programmable FPGAs each with:

- 18MB private DDR SDRAM
- 128 MB private DDR SDRAM

▶ 64-bit/66MHz CompactPCI

- 512MB Global DDR SDRAM
- Two PMC Sites with JN4 to FPGA
- External Data Port, up to 12Gb/s
- StarFabric PICMG2.17 Compliant Port



Quadia

Innovative Integration
... real time solutions!

www.innovative-dsp.com • 805.520.3300 phone

Designing For Signal Integrity

You can use the Xilinx/Ansoft 10 Gbps Backplane Design Kit to predict interconnect performance.

by Suresh Sivasubramaniam
Senior Design Engineer
Xilinx, Inc.
suresh.subramaniam@xilinx.com

Lisa Murphy
Application Engineer
Ansoft
lmurphy@ansoft.com

The Xilinx® Virtex-4™ FX family of devices contains up to 24 RocketIO™ multi-gigabit transceivers, each capable of operating anywhere from 622 Mbps to 11 Gbps. This seamless scalability, coupled with support for various emerging standards (Figure 1), allows you tremendous flexibility to upgrade today's designs to meet increasing bandwidth requirements.

To realize the full potential of this upgradeability to high-bandwidth processing applications, you must carefully design the serial interconnect channels on the PCB, be it line card or backplanes.

Once the transfer characteristics of the physical channel are well understood, you can effectively employ features such as transmit pre-emphasis/voltage swing and receive equalization (Figure 2) to overcome losses and attenuation in the channel, thus ensuring high signal integrity at the receiver.

MK322 Evaluation Board Case Study

The MK322 platform is the primary board used for the electrical evaluation and characterization of the RocketIO X high-speed serial multi-gigabit transceivers in Virtex-II Pro™ X FPGAs. This board was specifically designed to evaluate and test the RocketIO X transceiver and is available for sale.

The SMA connectors on the board allow you to interface the board to a scope, to other boards, or for loopback tests. The physical channel for each transceiver is carefully optimized to ensure the highest signal

quality at the SMAs (on the transmit path) or at the FPGA (on the receive path).

The data can significantly degrade after it has passed through the transmission path. Degradation includes loss of signal amplitude, reduction of signal rise time, and a spreading at the zero crossings. It is critical to model the transmission path when designing a high-performance, high-speed serial interconnect system. The transmission path may include long transmission lines, connectors, vias, and crosstalk from adjacent interconnect.

MK322 Board Stackup

The MK322 is a 12-layer board. The stack and trace geometries are designed for 100 Ohm differential and 50 Ohm single-ended signaling. The board material is standard FR4 ($\epsilon_r = 4.2$ and $\tan\delta = 0.02$). All trace and plane layers are 0.5 oz. copper (0.65 mil thick). The electrical channel of interest for our case study is routed as fol-

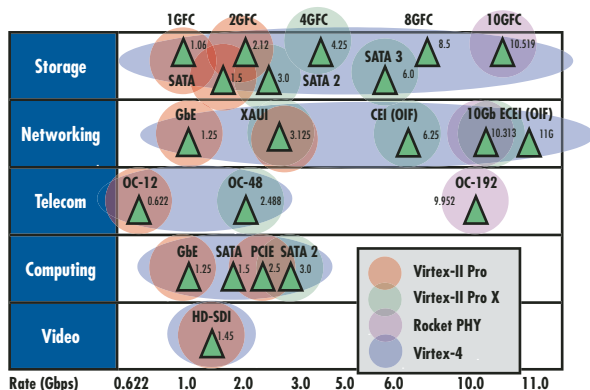


Figure 1 – Seamless scaling from 622 Mbps to 10 Gbps

Feature		Benefit
Programmable Termination	Yes	Reduces reflections
Programmable Voltage Swing	Yes	Reduces power
Transmit Pre-Emphasis	Yes	Equalizes simple channels
Integrated AC Coupling	Yes	Direct interface to other devices, reduces component count
Receive Equalization	Linear and DFE	Equalizes stringent channel; allows legacy backplanes to be upgraded
Automatic EQ Settings Algorithm	Yes	Automatically finds optimum EQ setting for a given channel; eases design and ensures signal integrity

Figure 2 – Programmable pre-emphasis and equalization features in the Virtex-4 FX family

lows: microstrip on the top layer and transitions to layer 10 stripline through a GSSG differential via.

Differential Signal Topology

The differential signals are routed into and out of the board using Rosenberger™ high-performance coax-to-board SMA connectors. The signals are routed from the top-mounted connector to the FPGA using stripline transmission lines (layer 10), which transition to microstrip before interfacing with the FPGA BGA package. The actual trace layout for one Tx and Rx pair is shown in Figure 3.

Modeling and Simulation

The electrical channel comprises five main sections (Figure 4):

- The BGA package
- Microstrip transmission line
- Differential via (GSSG configuration, G- ground, S- signal)
- Stripline transmission line
- Connector

Let's look at each piece in turn.

BGA Package

The package model and the specific Tx pair of interest were extracted from the Cadence™ APD database and simulated using Ansoft HFSS. Figure 5 is a plot of the differential insertion loss (red) and return loss (blue) as computed by Ansoft HFSS.

For this particular differential pair, return loss is better than 15 dB, up to 22 GHz. Ansoft HFSS can output the differential S-parameters as Touchstone files. Typically, companies are reluctant to give out their package databases except under an NDA, because they contain sensitive design information. However, you can use S-parameters derived from the model for channel simulations.

Microstrip and Stripline Interconnect

We performed simulations for the stripline and microstrip structures using the two-dimensional quasistatic finite element simulator within Ansoft SI 2D Extractor. The

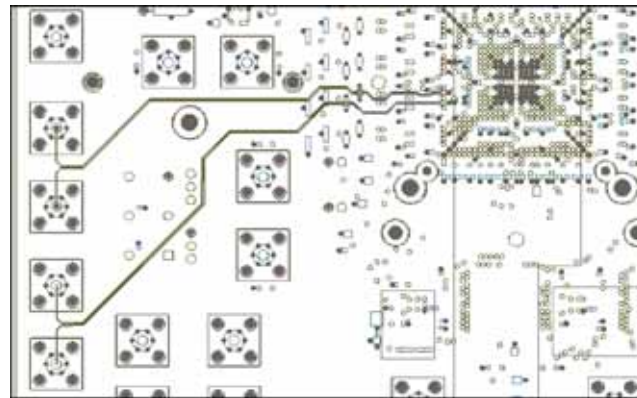


Figure 3 – Physical structure of a Tx and Rx differential pair on the MK322 board

stripline geometries were designed to provide nominally 100 Ohms differential impedance. Simulations confirmed that the impedance was within 7% of the nominal value (see Figure 6).

You can model PCB interconnects using various methods within Ansoft Designer™. The simplest is to use a coupled-line circuit model (like those found in popular high-frequency circuit simulators such as Ansoft Designer). In this instance, the interconnect is modeled with a uniform differential cou-

In addition, we parameterized each of the interconnect models. For example, in the microstrip interconnect model, the width, spacing, metal thickness, and physical length are parameters that can vary. For the initial simulations, these values were set to geometries specific to the MK322 board.

Differential Via

In keeping with good design practices that minimize unterminated stubs, layer 10 was used to transition from the microstrip

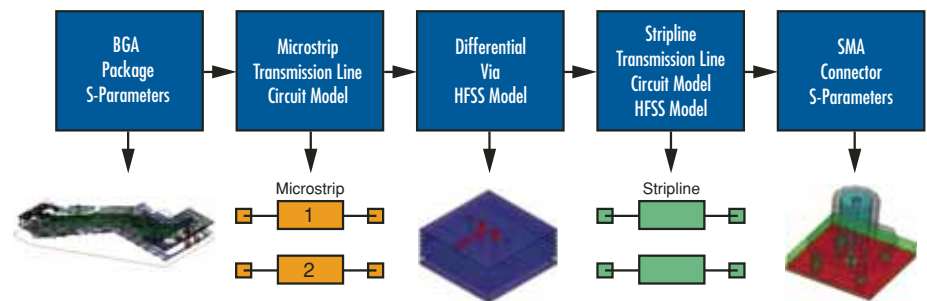


Figure 4 – The individual pieces comprising the full channel

pled transmission line without any discontinuities. On the other end of the modeling spectrum is the utilization of a full-wave planar EM field simulator based on the method of moments (MoM). Although accurate, MoM simulations are also the most computationally expensive method to predict interconnect performance.

A compromise that offers the accuracy of planar EM simulations with some of the speed of circuit simulation is offered by using a combination of the two. Figure 7

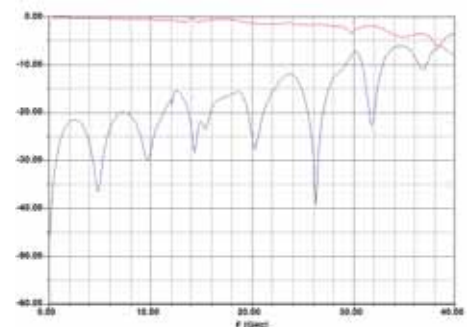


Figure 5 – Package model insertion loss (red) and return loss (blue) as computed by Ansoft HFSS

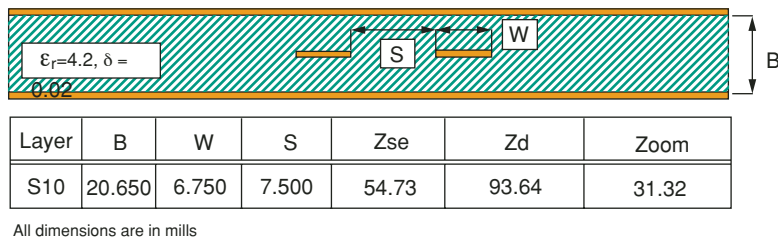


Figure 6 – Impedance for the stripline traces as extracted using Ansoft SI 2D Extractor

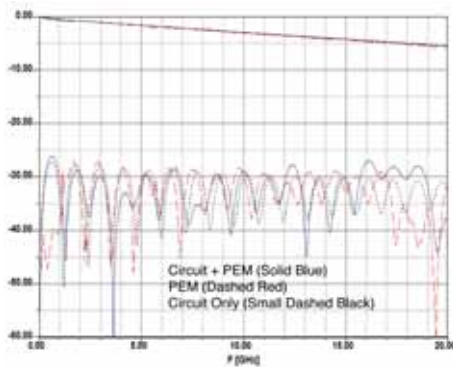


Figure 7 – A comparison of the three methods to simulate interconnects

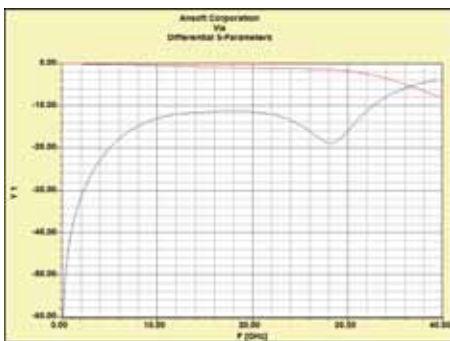


Figure 8 – Differential S-parameters for the via as computed by Ansoft HFSS

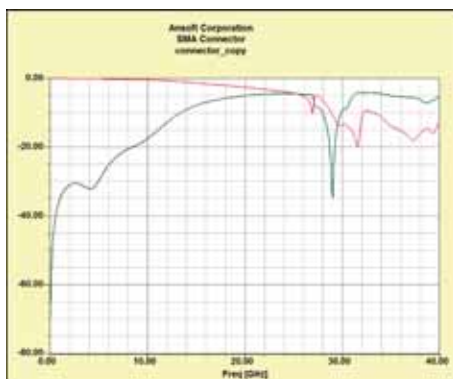


Figure 9 – Differential S-parameters for the SMA connector

to stripline using the throughhole differential via. The actual geometries for the ground-signal-signal-ground configuration were taken from Appendix D of the XFP specification (see pages 160-163 of the specification).

Several key variables for the via are parameterized, including spacing between signal vias, via radius, and antipad radius. Simulation results for the differential via structure are shown in Figure 8. The via structure shows excellent broadband insertion and return loss (> -10 dB) well beyond 20 GHz.

SMA Connector

The SMA connector used on the MK322 board is manufactured by Rosenberger (Part # 32K153-400). Rosenberger was gracious enough to provide us with the HFSS model for the connector, along with the optimized PCB footprint. The critical parameters for optimization involve the pad and antipad radii, as well as placement and spacing of several ground return vias around the center conductor. The ground vias around the center conductor allow the signal to transition from a radial coaxial field to a transverse electromagnetic mode (TEM) transmission line field in such a way that it minimizes any impedance mismatches. Figure 9 shows the insertion and return loss (> -10 dB up to 12 GHz) for the optimized SMA launch.

Full Channel Simulation

It is possible to cascade results generated from EM and circuit simulations on each

of the individual components to get a full system simulation. Figure 10 is a snapshot of the schematic of the full channel, from the SMA connector, through the board to the Xilinx Virtex-II Pro X BGA package, set up for frequency domain analysis.

Figure 11 is a plot of the system simulation results displaying the insertion and return loss up to 40 GHz. As expected, the channel has a response similar to a low-pass filter. The majority of the energy for a baseband digital binary signal is contained within the first null of its power spectrum. For the rise time and signaling rate of this channel (30 ps, 10 Gbps), we are most concerned with the response up to 17 GHz. As seen in the plot, the insertion loss is roughly -10 dB and the return loss is below -10 dB up to 17 GHz.

You can also perform time domain simulations (see Figure 12) using the system simulator in Ansoft Designer. This simulator uses a convolution algorithm to process the frequency domain channel data with user-defined input bitstreams. Insertion and return loss is included in the simulation.

An ideal 10 Gbps pseudo-random bit source with a 0.5V p-p amplitude and 30 ps rise time was applied to the channel.

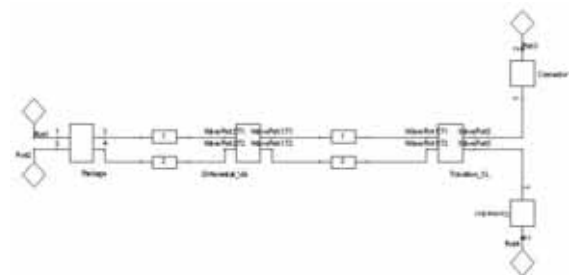


Figure 10 – Schematic of the full channel setup for frequency domain analysis within Ansoft Designer

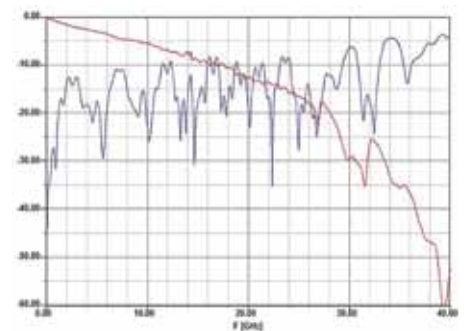


Figure 11 – Insertion and return loss for the full channel

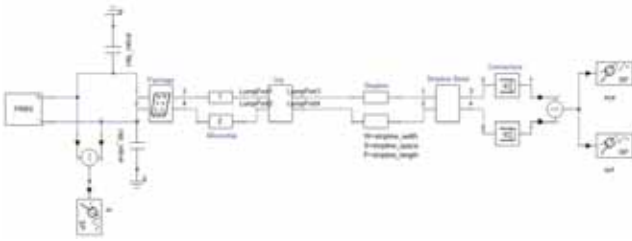


Figure 12 – Schematic showing setup for time-domain simulations

The channel was terminated in single-ended 50 Ohm impedances. The resulting eye diagram is shown in Figure 13, along with a measured eye diagram. There is excellent correlation between the measurement and simulation results. A very clear and open eye is achieved, as is expected from the frequency domain results.

For comparison to the measured eye, the driver capacitance was added to the channels. These capacitors are not part of the package model, because the passive channel will eventually be used with actual driver/receiver models that already include the capacitance. No pre-emphasis was used in the simulation. It should be anticipated that some pre-emphasis would sharpen up the time-domain response.

Extension of the Methodology

In creating the models, we emphasized that the critical variables that make up the physical structure are parameterized. Why parameterize? Although there are many reasons for doing so, let's show through some examples the power and utility of models that allow manipulation of critical variables.

A Longer Stripline Segment

In the original model, the nominal length for the stripline segment of the channel is 2.5 in. For whatever reason (board routing congestion is an obvious one), suppose that the stripline segment now needed to be 5 in. You can easily investigate the channel performance for this new scenario by changing the physical length variable (SL_L) in the model. Examples of such an analysis, for various trace lengths, are shown in Figure 14.

Increasing the length of the stripline segments results in significant eye degradation. Because every component of the

channel is parameterized, you can explore the performance impact of different variables in each section of the channel when investigating design trade-offs. In fact, with exactly this intent in mind, we have made

these models available as a Xilinx/Ansoft 10 Gbps Backplane Design Kit at www.gigabitbackplanedesign.com. Complete details on each of the models and the parameterized variables are available at this site.

Conclusion

Modern platform FPGA devices provide wide bandwidth processing and high-speed I/O. Serial I/O with speeds in the gigabit realm creates new challenges for PCB designers.

Models associated with this effort have been assembled into a 10 Gbps backplane design kit that you can use to predict performance of circuit board designs.

The design kit is available on the Xilinx "SI Central" website, enabling you to rapidly evaluate your own board designs. Visit www.gigabitbackplanedesign.com for more information.

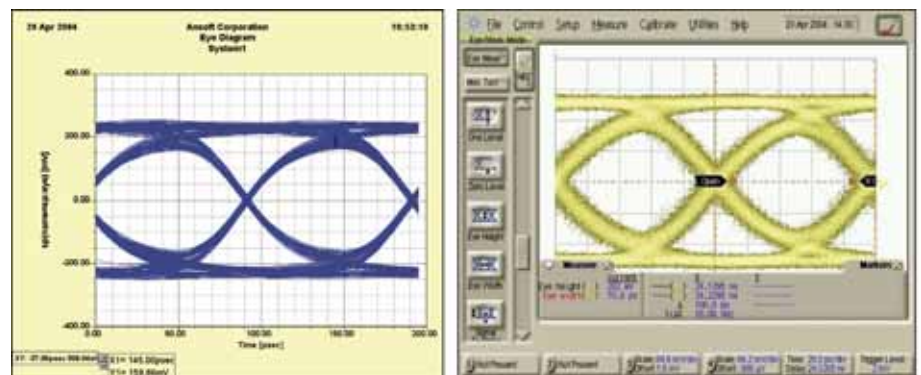


Figure 13 – Simulated (left) and measured (right) eye diagram for the full channel; the simulated eye is in excellent agreement with measurements

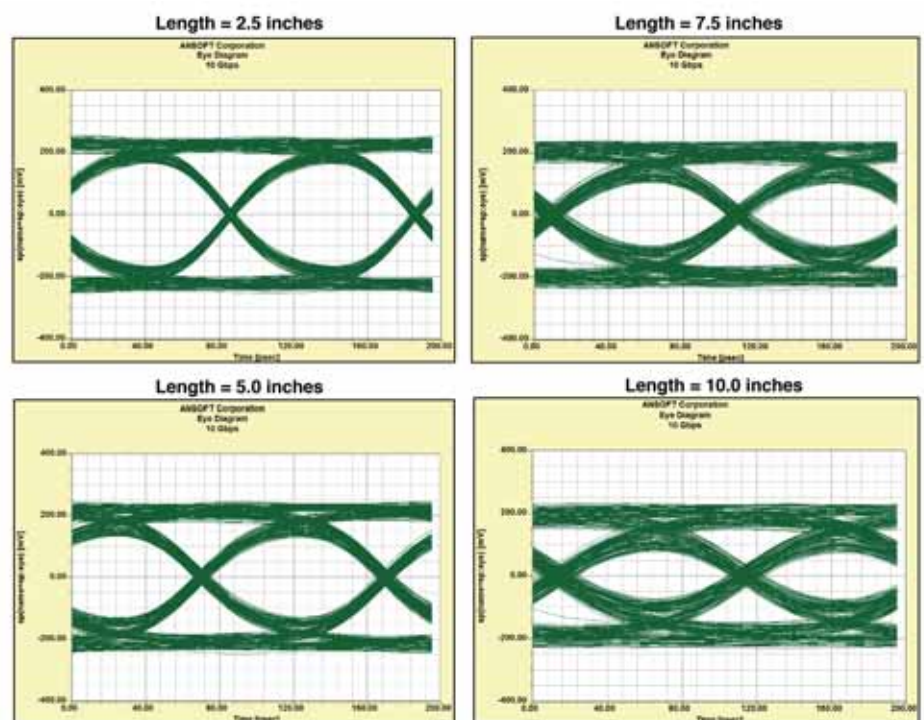


Figure 14 – Channel performance degrades due to losses in the transmission line as the trace length increases

Accelerated System Performance with APU-Enhanced Processing

The Auxiliary Processor Unit (APU) controller is a key embedded processing feature in the Virtex-4 FX family.

by Ahmad Ansari
Senior Staff Systems Architect
Xilinx, Inc.
ahmad.ansari@xilinx.com

Peter Ryser
Manager, Systems Engineering
Xilinx, Inc.
peter.ryser@xilinx.com

Dan Isaacs
Director, APD Embedded Marketing
Xilinx, Inc.
dan.isaacs@xilinx.com

The APU controller provides a flexible high-bandwidth interface between the re-configurable logic in the FPGA fabric and the pipeline of the integrated IBM™ PowerPC™ 405 CPU. Fabric co-processor modules (FCM) implemented in the FPGA fabric are connected to the embedded PowerPC processor through the APU controller interface to enable user-defined configurable hardware accelerators. These hardware accelerator functions operate as extensions to the PowerPC 405, thereby offloading the CPU from demanding computational tasks.

APU Instructions

The APU controller allows you to extend the native PowerPC 405 instruction set with custom instructions that are executed by the soft

FCM; the primary capabilities are shown in Figure 1. This provides a more efficient integration between an application-specific function and the processor pipeline than is possible using a memory-mapped coprocessor and shared bus implementation.

The instructions supported by the APU are classified into three main categories:

- User-defined instructions (UDI)
- PowerPC floating-point instructions
- APU load/store instructions

The UDIs are programmed into the controller either dynamically through the PowerPC 405 device control register (DCR) or statically when the FPGA is configured through its bitstream. The APU controller allows you to optimize your system architecture by decoding instructions either internally or in the FCM.

The floating-point unit (FPU) is an example of an FCM. The PowerPC floating-point instruction set is decoded in the APU controller, whereas the computational functionality is implemented in the FPGA fabric. To support FPUs with different complexities, the APU controller allows you to select subgroups of the PowerPC floating-point instructions. These instructions are executed in the FCM while other subgroups of instructions are either computed through software FPU

emulation or ignored completely. This fine-tuning optimizes FPGA resources while accelerating the most critical calculations with dedicated logic.

The APU controller also decodes high-performance load and store instructions between the processor data cache or system memory and the FPGA fabric. A single instruction transfers up to 16 bytes of data – four times greater than a load or store instruction for one of the general purpose registers (GPR) in the processor itself. Thus, this capability creates a low-latency and high-bandwidth data path to and from the FCM.

APU Controller Operation

Figure 2 identifies the key modules of the APU controller and the 405 CPU in relation to the FCM soft coprocessor module implemented in FPGA logic. To explain the operation of the APU controller and the processor interactions related to the execution units in soft logic, we can trace the step-by-step sequence of events that occur when an instruction is fetched from cache or memory.

Once the instruction reaches the decode stage, it is simultaneously presented to both the CPU and APU decode blocks. If the instruction is detected as a CPU instruction, the CPU will continue to execute the instruction as it would normally. Otherwise, within the same cycle, the CPU

- **Extends PPC 405 Instruction Set**
 - Floating Point Support (with soft auxiliary processor)
 - User-Defined Instructions
- **Offloads CPU-Intensive Operations**
 - Matrix Calculations
 - Video Processing
 - Floating-Point Mathematics
 - 3D Data Processing
- **Direct Interface to HW Accelerators**
 - High Bandwidth
 - Low Latency

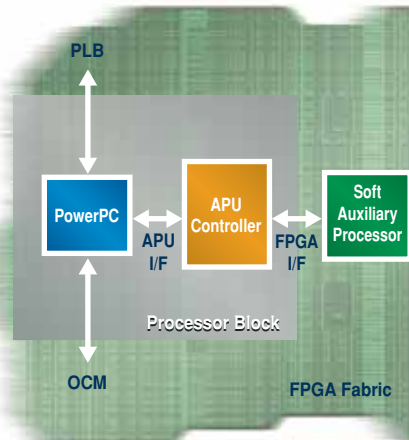


Figure 1 – APU expanded processing capabilities

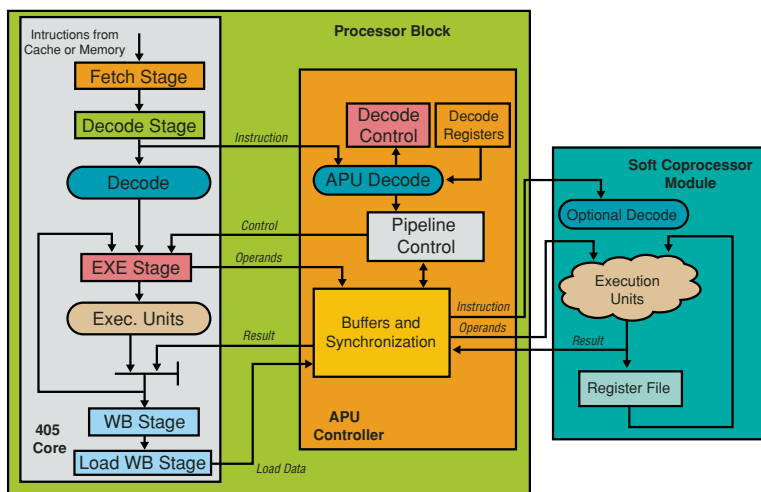


Figure 2 – APU controller processing operative block diagram

will look for a response from the APU controller. If the APU controller recognizes the instruction, it will provide the necessary information back to the CPU.

If the APU controller does not respond within that same cycle, an invalid instruction exception will be generated by the CPU. If the instruction is a valid and recognized instruction, the necessary operands are fetched from the processor and passed to the FCM for processing.

Because the PowerPC processor and the FCM reside in two separate clock domains, synchronization modules of the APU controller manage the clock frequency difference. This allows the FCM to operate at a slower frequency than the processor. In this instance, the APU controller would receive the resultant data from the coprocessor and

at the proper execution time send the data back to the processor. The APU controller knows in advance, based on instruction type, if or when it will get the result.

Autonomous and Non-Autonomous Instructions

Two major categories of instructions exist: autonomous and non-autonomous. For autonomous instructions, the CPU continues issuing instructions and does not stall while the FCM is operating on an instruction. This overlap of execution allows you to achieve high performance through techniques such as software pipelining.

On the other hand, during the synchronized execution, the CPU pipeline stalls while the FCM is operating on an instruction. This feature allows you to

implement synchronization semantics to pace the software execution with the hardware FCM latency.

Non-autonomous instruction types are further divided into blocking and non-blocking. If blocking, asynchronous exceptions or interrupts are blocked until the FCM instruction completes. Otherwise, if non-blocking, the exception or interrupt is taken and the FCM is flushed.

Software Description

Software engineers can access the FCM from within assembler or C code. On one side, Xilinx has enabled the GCC compiler (which is contained in the Embedded Development Kit) to generate code that uses an FCM floating-point unit to calculate floating-point operations. Furthermore, assembler mnemonics are available for UDIs and the pre-defined load/store instructions, enabling you to place hardware-accelerated functions into the regular program flow. For the ultimate level of flexibility, you can define your own instructions designed specifically for the hardware functionality of the FCM.

You can easily use the pre-defined load/store instructions through high-level C macros. For example, in an application where the FCM is used to convert pixel data into the frequency domain, 8 pixels of 16 bits are transferred from main memory to an FCM register with a simple program:

```
unsigned short pixel_row[8]; // 8 pixels,
                             // each pixel has a size of 16 bits
lqfcm(0, pixel_row); // transfer a row of
                     // pixels to FCM register zero
```

The quadword load operation maintains cache coherency as the data is moved through the cache, if caching is enabled for the corresponding address space.

The FCM operation on the pixel data can start on an explicit command; for example, a UDI. However, for many applications the operation starts immediately after the FCM hardware detects the completion of the load instruction.

The latter approach has many advantages:

- Simple software – A load operation moves the data from the memory to

the FCM and starts the operation. A subsequent store instruction retrieves the result of the operation and stores it back to main memory.

- High data transfer rates – Quadword load and store operations take just a few cycles to complete. A single operation moves 16 bytes within that timeframe.
- Low latency – FCM load operations are simple to use. The processor completes the operation in a single cycle.

The principle of the RISC architecture uses a number of simple instructions on data stored in general-purpose registers (GPR) to compute complex operations. User-defined instructions fall into this category but take the concept a step further in that the system architect defines the complexity of the operation on data stored in GPRs and FCM registers (FCR). Again, from a software point of view, the engineer codes user-defined instructions through C macros. GCC recognizes mnemonics such as `udi0fcm` as a user-defined operation of the general form:

```
udi0fcm<FCRT5/RT5>,<FCRA5/RA5/imm>,<FCRB5/RB5/imm>
```

The target of the operation is either a GPR or an FCR. The operands are either GPRs, FCRs, immediate values, or a combination. As you can see, the semantics are not defined by the instruction and depend on your intentions and the implementation in the FCM.

This code sequence demonstrates the use of a user-defined instruction as an example of a complex add operation:

```
struct complex {
    int r, i;    // 32 bit integer for real
               // and imaginary parts
};
complex a, b, r;
ldfcm(0, &a);  // load complex number a
into FCM register 0
ldfcm(1, &b);  // load complex number b
into FCM register 1
udi0fcm(2, 1, 0); // udi0fcm computes r = a
+ b, where r is stored in FCM register 2
stdfcm(&r, 2);  // store complex result
from FCM register 2 to variable r
```

To increase the readability of the code, you can redefine the user-defined instruction with regular C preprocessor constructs. Instead of using the `udi0fcm()` macro, you can redefine it to a more comprehensible `complex_add()` macro with `#define complex_add(r, a, b) udi0fcm(r, a, b)` and change the listing to call `complex_add(2, 1, 0)` instead of `udi0fcm(2, 1, 0)`.

Therefore, system architects can partition their tasks into hardware- and software-executed pieces that are efficiently and precisely interfaced to one another through the use of the APU controller. This partitioning can be done statically during the initial system configuration or dynamically during the program execution. Using the direct processor/FPGA coupling presented by the APU controller and its high throughput interfaces, hardware/software synchronization is greatly simplified and performance significantly improved.

Accelerating System Performance

The following examples showcase key advantages the APU provides based on two different scenarios. The first scenario is essentially a benchmarking comparison of a finite impulse response (FIR) filter using a soft FPU core, implemented as an FCM attached directly to the APU controller (as compared to software emulation used to calculate the filter function). The second scenario implements a two-dimensional

inverse discrete cosine transform (2D-IDCT) typically used as one of the processing blocks in MPEG-2 video decompression, again compared to emulating the 2D-IDCT function in software.

The two use cases are different in that the FPU implements a set of registers in the FPGA fabric upon which the FPU instructions operate. The 2D-IDCT only requires load and store operations, while the functionality of the operation on the data stream is fixed. In either case the operations are complex enough to justify offloading into the FPGA fabric.

Thus, the combination of using the APU and FPGA hardware acceleration clearly provides a significant performance advantage over software emulation – or the conventional method involving the processor and processor local bus architecture with a soft co-processing function.

FIR Filter

The implementation of floating-point calculations in hardware yields an improvement by a factor of 20 over software emulation. Connecting the FPU as an FCM to the APU controller provides performance improvement because the latency to access the floating-point registers is reduced and dedicated load and store instructions move the operands and results between the FPU registers and the system memory.

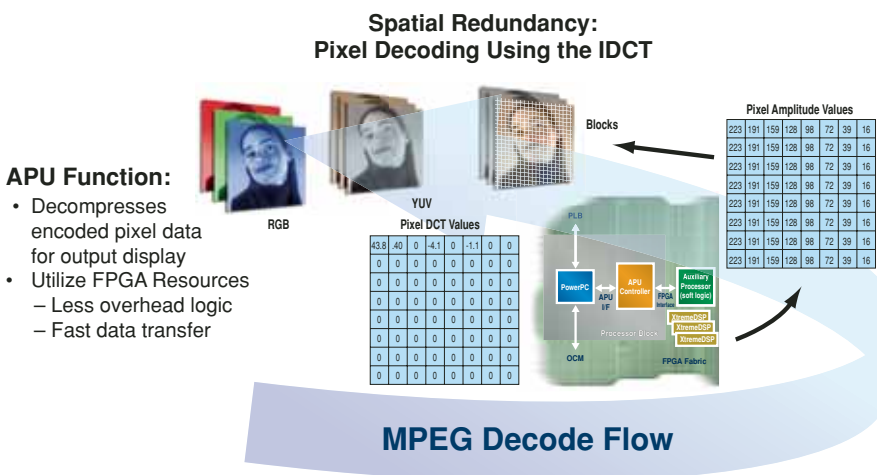
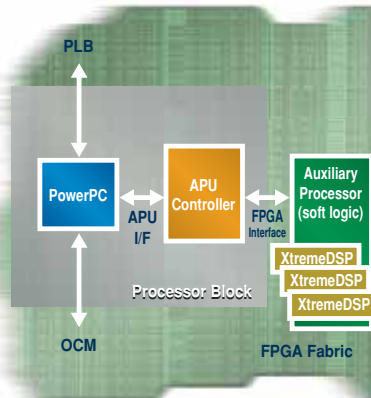


Figure 3 – Utilizing APU to decode pixel data for display output

Example: Video Application – MPEG De-Compression Algorithm

- **Leverages Integrated Features**
 - PowerPC, APU, XtremeDSP Blocks
- **HW Acceleration Over Software**
 - Lower Latency and High Bandwidth
- **Efficient HW/SW Design Partitioning**
 - Optimized Implementation
- **Significant Performance Increase**



Over 20X Performance Improvement Compared to Software Emulation

Figure 4 – Accelerated system performance with APU

2D-IDCT

The 2D-IDCT transforms a block of 8 x 8 data points from the frequency domain into pixel information. A high-level diagram depicting the pixel decode by the APU controller, along with advantages, is shown in Figure 3. In this example, each data point has a resolution of 12 bits and is represented as a 16-bit integer value. The data structure is defined where each row of 8 pixels consumes 16 bytes. This is an ideal size that allows optimal use of the FCM load and store instructions described earlier. In other words, eight FCM quadword load instructions are needed to load a data block into the 2D-IDCT hardware. Eight FCM quadword store instructions are sufficient to copy the pixel data back into the system memory.

The calculation of the 2D-IDCT in the FCM starts immediately after the first load, and the pixel data is available shortly after the last load operation. As shown in Figure 4, the 2D-IDCT makes use of the new XtremeDSP™ slices in the Virtex-4 architecture that offer multiply-and-accumulate functionality.

A software-only implementation of a 2D-IDCT takes 11 multiplies and 29 additions together with a number of 32-bit load and store operations, while the hardware-accelerated version takes 8 load and 8 store operations. The reduced number of operations results in a speed-up of 20X in favor of a 2D-IDCT FCM attached through the APU controller.

By comparison, if you connect the 2D-IDCT hardware block to the processor local bus, as it is done conventionally, the system performance will be reduced. This increased latency is mainly caused by the bus arbitration overhead and the large number of 32-bit load and store instructions. This is illustrated schematically in Figure 5.

Conclusion

The low-latency and high-bandwidth fabric coprocessor module interface of the APU controller enables you to accelerate algorithms through the use of dedicated hardware. Where operations are complex enough to justify the offloading into the FPGA fabric, or when acceleration of a

specific algorithm is desired to achieve optimal performance, the combination of the APU controller and FPGA hardware acceleration provides a definitive performance advantage over software emulation or the conventional method of attaching coprocessors to the processor memory bus.

Generating the accelerated functions called by user-defined instructions is easily performed through GUI-based wizards. This functionality will be included in subsequent releases of the powerful Embedded Development Kit or Platform Studio.

If you are more comfortable working at the source code or assembly level, the APU controller allows you to define your own instructions written specifically for the hardware functionality of the FCM, or you can easily use the pre-defined load/store instructions through high-level C macros.

The APU controller provides a close coupling between the PowerPC processor and the FPGA fabric. This opens up an entire range of applications that can immediately benefit customers by achieving increases in system performance that were previously unattainable.

For additional details on the APU controller in Virtex-4-FX devices, including detailed descriptions and timing waveforms, refer to the Virtex-4 PowerPC 405 Processor Block Reference Guide at www.xilinx.com/bvdocs/userguides/ug018.pdf. 

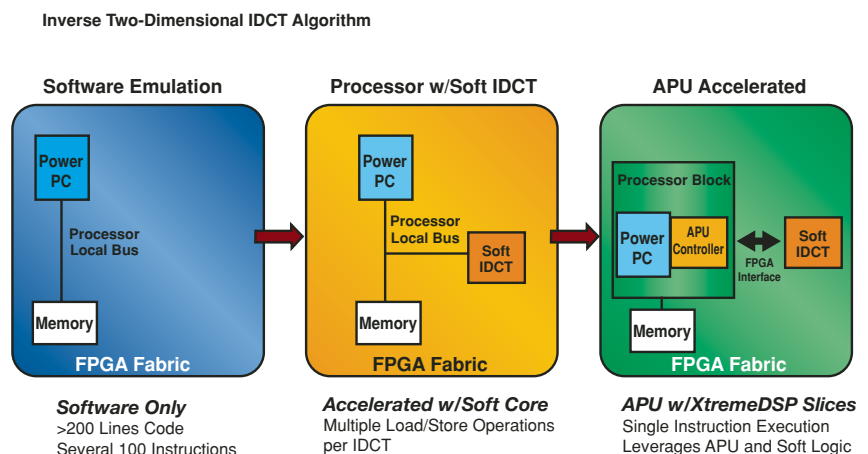


Figure 5 – Comparison of implementation models for 2D-IDCT

Solving the Signal Integrity Challenge

Virtex-4 RocketIO transceivers bring blazing speed, and the ability to use it.

by Ryan Carlson
Director of Marketing, High Speed Serial I/O
Xilinx, Inc.
ryan.carlson@xilinx.com

The industry is moving away from parallel buses and relatively slow differential signals toward higher speed differential signaling schemes. These high-speed signals solve many design challenges: they offer new levels of bandwidth, they lower overall system cost, and they make designs easier by addressing the skew issues of large parallel buses.

However, with these improvements comes a new challenge: maintaining signal integrity. As signals push the limits of the media across which they are transmitted, the challenge of dealing with signal impairments becomes non-trivial, to say the least. The new Xilinx® Virtex-4™ RocketIO™ transceivers have incorporated multiple new features designed to solve this challenge.

Frequency-Dependent Loss

Several factors contribute to the frequency-dependent loss of a typical channel. Figure 1 shows the frequency response of 1 m of FR-4 trace. Dielectric loss and skin effect combine to create a significant loss above 1 GHz. With today's serial I/O standards

approaching 10 Gbps, this loss becomes a critical design issue.

As a signal travels across a channel (like the one with a transfer function shown in Figure 1), a bit is degraded to the point where it interferes with neighboring bits; this is known as inter-symbol interference (ISI). Figure 2 shows the effect of ISI on a signal transmitted across a typical back-plane channel. The high-frequency components are subject to losses that are greater than the low-frequency components. The edges that contain the high-frequency components are degraded, resulting in added jitter and eye closure. Additional techniques are needed to compensate for these losses.

Signal Integrity Features

The Virtex-4 RocketIO transceivers contain several features aimed at solving this problem. The first is transmit pre-emphasis. By modifying the signal before it is transmitted through a channel, transmit pre-emphasis can proactively compensate for some of the frequency-dependent loss of the channel.

Although most existing solutions use two-tap transmit pre-emphasis (addressing only the post-cursor ISI shown in Figure 2),

the Virtex-4 RocketIO transceivers employ three-tap transmit pre-emphasis to address both pre- and post-cursor ISI. For signal rates above 3 Gbps, pre-cursor ISI becomes a non-negligible effect, and three taps of transmit pre-emphasis are needed to solve the problem.

In addition to transmit pre-emphasis, Virtex-4 RocketIO transceivers provide two different types of receive equalization. These options can be used in conjunction with transmit pre-emphasis to further improve signals degraded by lossy channels.

The first type of receive equalization works by amplifying the high-frequency components of the signal that have been attenuated by the channel (Figure 1). The transfer functions of this equalizer are programmable, and are shown in Figure 3.

The second type of receive equalization is called decision feedback equalization (DFE). This technique removes ISI effects by looking at consecutive bits and choosing the amount of equalization needed.

Both forms of receive equalization described above seek to amplify the high-frequency components of the desired signal. An advantage of DFE is that it does not amplify any crosstalk that may be associated with the signal. This technique can

therefore be useful for increasing the speed of legacy backplanes, where extensive crosstalk may exist.

All of these signal integrity features are fully programmable; they can be used independently or together, and each has multiple settings to equalize any channel. To fully take advantage of these hardware-based features, Xilinx also provides software-based reference designs that use bit error rate tests (BERT) to find the optimal settings for each unique application.

Integrated Receive Side AC-Coupling Capacitors

Many applications require AC-coupling capacitors to ensure compatibility between different Tx and Rx blocks. These capacitors require their own vias; at high speeds vias present yet another discontinuity to impair signal quality.

The Virtex-4 RocketIO transceivers integrate the AC-coupling capacitors on chip. This not only reduces external component count and design effort, but more importantly improves signal integrity by removing the need for extra vias in the board. These integrated AC-coupling capacitors can be optionally bypassed.

Conclusion

Signal integrity is an engineering challenge that accompanies the move to high-speed serial signaling. Once the system design has been optimized to minimize the physical effects of connectors, board materials, traces, vias, coupling capacitors, and cables, the remaining losses and channel effects need to be addressed by advanced silicon features.

Virtex-4 RocketIO transceivers are the industry's fastest integrated transceivers. Along with these leading-edge speeds, the RocketIO transceivers deliver multiple features designed to simultaneously address the signal integrity challenge that comes with them.

Xilinx has detailed information about high-speed design challenges, and the solutions available to solve them, at www.xilinx.com/signalintegrity. Instructional DVDs that describe various aspects of the signal integrity challenge can be purchased from the Xilinx online store by visiting www.xilinx.com/store/.

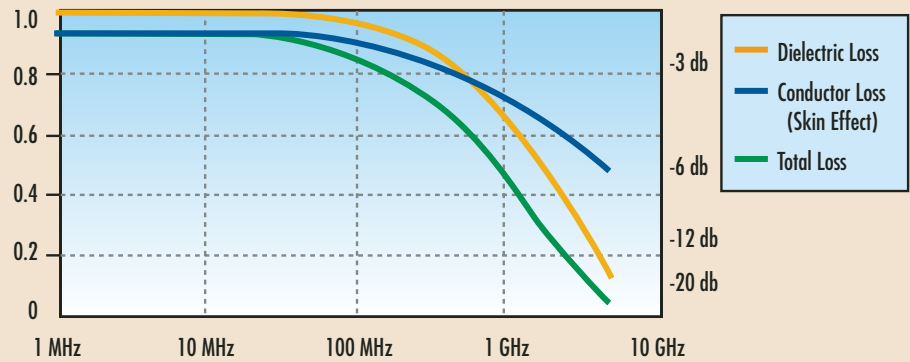


Figure 1 – Frequency-dependent loss

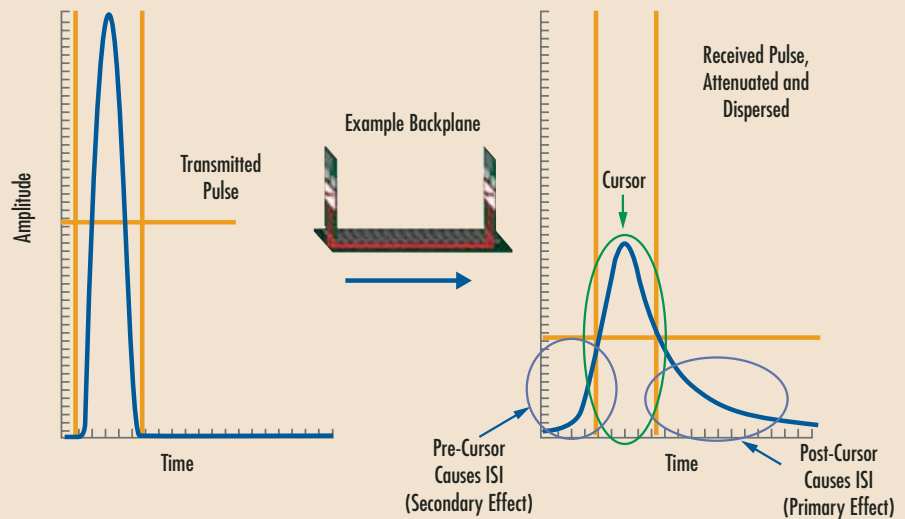


Figure 2 – A transmitted bit (left) and the result of inter-symbol interference (right)

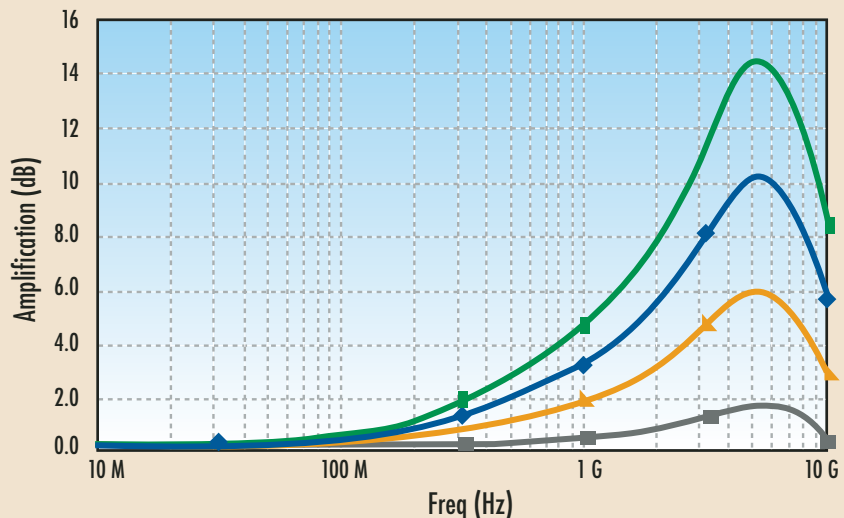


Figure 3 – Virtex-4 RocketIO receive equalization transfer functions



Using FPGAs in Wireless Base Station Designs

Wireless base station design trends benefit from Virtex-4 device features.

by David Gamba

Senior Manager, Strategic Solutions Marketing
Xilinx, Inc.

david.gamba@xilinx.com

Wireless infrastructure revenue continues to experience phenomenal growth, increasing from approximately \$27 billion in 2003 to an estimated \$35 billion in 2004. Industry analysts are predicting that 2004 will be the peak revenue year, as forecasts show the revenue figure dropping back to \$27 billion in 2005, eventually settling in to the \$10-\$15 billion range by the end of the decade. This revenue decline is driven both by lower prices as well as a drop in base station deployments, from nearly 500,000 stations in 2004 to less than 200,000 in 2010.

As the industry transitions from a high-growth phase to a more mature state, cost pressures will increasingly mount in all facets of the infrastructure, including the wireless base station. Next-generation base station deployments must conquer the challenge of continually reducing cost (as measured by cost per channel) while adding functionality to support new services, protocols, and changing subscriber usage patterns.

To begin addressing this challenge, wireless base station designs are shifting from ASIC technology to more readily available off-the-shelf components such as FPGAs. This shift is driven both by declining annual base station unit volumes as well as FPGA technology improvements that increase processing power and enable a much lower cost per channel.

The migration to FPGAs is not just an attempt to reduce costs and create a common platform to achieve commoditization – it is also being driven by time-to-market pressures, along with the need to make in-

more manageable, avoiding some of the multi-million dollar inventory obsolescence issues that base station manufacturers have faced with ASIC solutions fabricated to support the 3G launch.

Standardizing the Wireless Base Station

Another significant step taken by the wireless industry is the launch of industry organizations focused on standardizing the non-differentiated features inside a base station. The most notable development for Xilinx is the migration to a standardized high-speed serial interconnect solution

Extending Current Design Lifecycles

Standardization is the first step towards the commoditization of base station design and will eventually lead to a phasing out of ASICs from wireless base stations. In the interim, companies are inserting discrete devices next to their current ASICs to support new functionality that cannot be added in a timely or cost-effective manner to the current design.

For instance, the Third Generation Partnership Project (3GPP), which is a collaboration agreement between several telecommunications bodies, is actively creating additional standards for the wireless industry. 3GPP has added a high-speed downlink packet access (HSDPA) feature as a new Universal Mobile Telecommunications System (UMTS) requirement in its latest baseband processing specification, Release 5, for Wideband Code Division Multiple Access (W-CDMA).

ASICs in current base stations do not support this new variant for UMTS. This creates a hole in the service offerings for UMTS, which forecasters are predicting will represent approximately 80% of the wireless traffic in the next few years. This deficiency must be addressed before future field deployments, and it can be – without exceeding the system power budget – by using a Virtex-4 LX device next to the ASIC, implementing HSDPA using the available Xilinx HSDPA IP offering.

Next-Generation Base Station Designs

But adding external devices to patch design holes created by existing ASIC designs limitations is purely a stopgap solution. Future base station designs must be able to quickly adapt to changes in subscriber traffic patterns, as well as support the upcoming convergence of new services and emerging cellular technologies such as W-CDMA, TD-SCDMA, EDGE, 1xEV-DO, and WiMAX.

As shown in Figure 2, the amount of cellular technologies is expected to continue to proliferate, leading base stations down the path of having to support many more technologies. Current issues such as

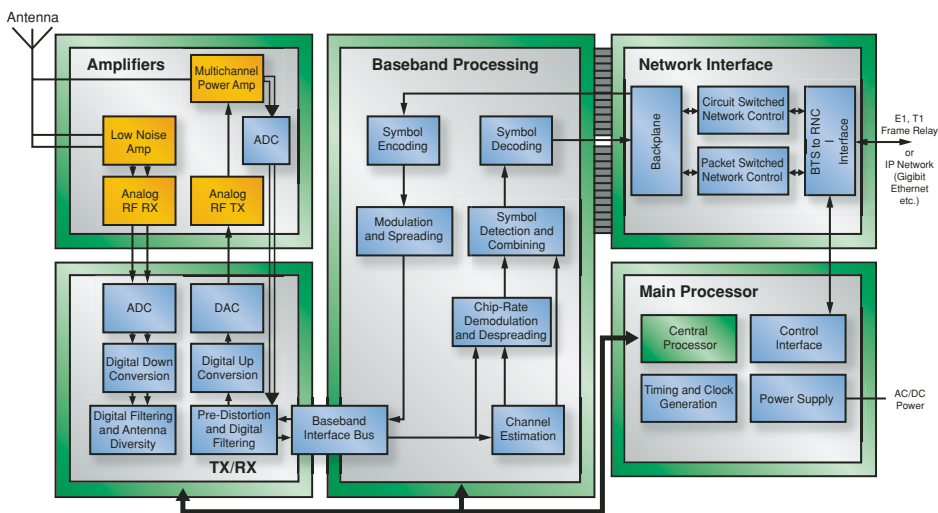


Figure 1 – Wireless base station module block diagram

field upgrades of base station deployments. This shift away from ASICs has enabled significant new design opportunities for Xilinx® Virtex-4™ devices to fill the void.

Wireless Base Station Module Building Blocks

Inside a wireless base station are fairly distinct module blocks performing different functions, such as radio, baseband processing, transport network interfacing, and control (Figure 1). Traditional base station designs used ASICs – along with DSPs and other discrete components – to implement these various architectural features and functions.

This design approach is rapidly giving way to more cost-effective and flexible designs that use FPGAs. With lower costs and increased flexibility, product delivery is accelerated and inventory control is much

between the different base station module blocks, such as the Open Base Station Architecture Initiative (OBSAI) Reference Point 3 (RP3) and Common Public Radio Interface (CPRI) interconnects for baseband and radio module connectivity.

Many leading base station manufacturers are members of these organizations and are rapidly preparing to adopt one of these two standard interconnect solutions in their upcoming design implementations. Xilinx is fully prepared to support these standards, and has both OBSAI and CPRI IP solutions and reference designs available for implementing in Virtex-II Pro™, Virtex-II Pro X, and Virtex-4 FX FPGA devices, using the integrated RocketIO™ multi-gigabit transceivers (MGTs) in association with the logic building blocks.

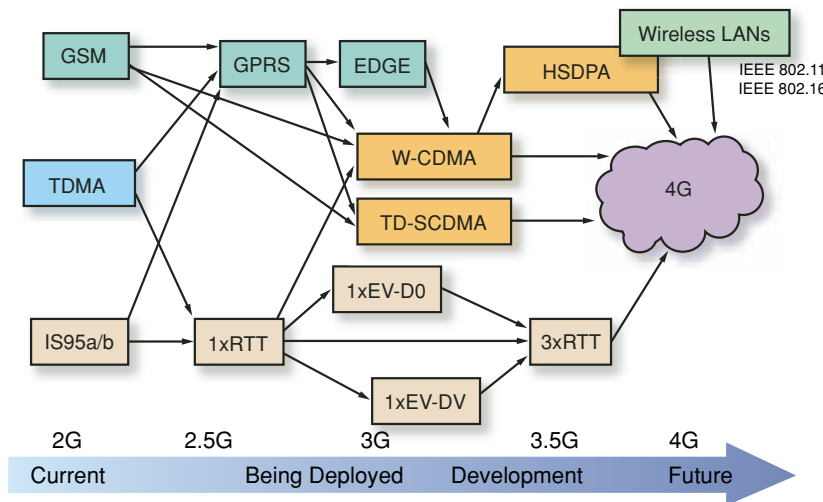


Figure 2 – Mobile technology roadmap

multi-user detection and antenna selection will be augmented by new technical challenges, such as channel provisioning and base station tuning, that will need to be resolved appropriately to reduce a service provider's customer turnover. The fundamental expectation to receive the same high-quality wireless service wherever a customer roams must be completely addressed.

These customer expectations would benefit from substantial flexibility in the base station. Fortunately, many of the baseband processing functions and radio module functions are well suited for implementation in Virtex-4 devices, taking

advantage of the integrated XtremeDSP™ slices in the product architecture.

For instance, quite a few baseband processing tasks – such as call initiation and set-up and multi-path signal detection and monitoring – are heavily based on mathematical algorithms. You can very efficiently implement these algorithms by using the integrated multiplier capabilities available in Virtex-4 devices, along with the readily available intellectual property components such as the Random Access Channel (RACH), Searcher, and 3G Turbo Convolutional Codecs (3GTCC) that Xilinx has imple-

mented as reference designs to demonstrate these capabilities.

The integrated DSP capability in the Virtex-4 SX device enables a very low power implementation of these functions. Radio functions can be expanded by using a Virtex-4 SX device to enable more channel support.

Several enabling pieces of intellectual property targeted at radio functions, such as digital pre-distortion (DPD), crest factor reduction (CFR), and digital up/down conversion (DUC/DDC), are supported by the Virtex-4 SX device. Not only does this help increase in the number of channels supported in a base station, but it also helps reduce the cost per channel. Table 1 gives an overview of the different capabilities offered by Xilinx baseband and radio module IP offerings.

System Generator for DSP Development Tool

Xilinx complements its Virtex-4 product offerings with the System Generator for DSP tool. This is a complete integrated DSP design environment that simplifies the development, debug, and verification of high-performance DSP designs targeting wireless base stations. This tool also helps designers interface with complementary general-purpose and DSP processors used in wireless base station designs.

System Generator for DSP provides high-level abstractions that are automatically compiled into Virtex-4 devices at the push of a button, with no loss in performance over designs implemented in lower-level languages such as VHDL. System Generator is part of the XtremeDSP solution, which combines state-of-the-art FPGAs, design tools, intellectual property cores, and design and education services.

Conclusion

To learn more about the key markets and end applications of Xilinx wireless solutions, visit www.xilinx.com/espl/, or e-mail 3g@xilinx.com. For more details about Virtex-4 FPGAs, visit www.xilinx.com/virtex4/. And for more details on System Generator for DSP or other pieces of the Xilinx DSP solution, visit www.xilinx.com/dspl/.

Xilinx Baseband Intellectual Property Offerings	
IP Offering	Application
HSDPA	Increases downlink data transmission rate to a peak of 14.4 Mbps
RACH	Receiver path preamble detection (specified by W-CDMA)
Searcher	Multi-path delay estimate for each subscriber
3G TCC	Forward error correction
Xilinx Radio Intellectual Property Offerings	
IP Offering	Application
DPD	Signal conditioning to enable use of lower cost RF power amplifiers
CFR	Signal amplitude conditioning to enable increased RF power amplifier efficiency
DUC	Baseband signal modulation for digital-to-analog converter input
DDC	Receiver signal modulation for analog-to-digital converter input

Table 1 – Xilinx baseband and radio IP offerings

Over **100 Million** SERVED.



A new benchmark in delivery!

The supremely popular, low-cost Spartan™ FPGA product line from Xilinx recently shipped its 100 millionth device. And we are in high-volume production of our 90nm Spartan-3 series, already delivered to customers worldwide. Addressing the demands of consumer-oriented, cost-sensitive applications, Spartan-3 FPGAs offer full-feature capability with the lowest price points ever.

Get started today with the world's lowest-cost FPGA

The Spartan-3 Starter Kit gives you instant access to the FPGA's complete platform capabilities, bringing high-volume designs to reality faster. The kit includes a total starter board, JTAG cable, handbook and resource CD, plus free ISE software, all for just US \$99. Contact your local distributor, or order your Spartan-3 Starter Kit today at www.xilinx.com/spartan3.

Now there's a hundred million reasons to get started today!



→ **MAKE IT YOUR ASIC**



The Programmable Logic CompanySM

www.xilinx.com/spartan3



©2004 Xilinx, Inc., 2100 Logic Drive, San Jose, CA 95124. Europe +44-870-7350-600; Japan +81-3-5321-7711; Asia Pacific +852-2-424-5200; Xilinx is a registered trademark, Spartan is a trademark, and The Programmable Logic Company is a service mark of Xilinx, Inc.

Proven 90nm Process
Lowest Cost FPGAs





Implementing a Cable Modem Termination System with Virtex-4 FPGAs

Integrated features make the Virtex-4 device an ideal choice.

by Delfin Rodillas
Strategic Solutions Manager
Xilinx, Inc.
delfin.rodillas@xilinx.com

With the continued proliferation of cable and satellite television and the rapid growth of the Internet, video transmission bandwidth has experienced phenomenal growth. With video streaming now being introduced into mobile handsets, this growth rate is not showing any signs of slowing down.

The technology advances of Xilinx® FPGAs have kept pace with the increasing transmission requirements and have solved many of the critical design issues in these systems. The Virtex-4™ product family incorporates additional enhancements – high-speed DSP, ultra low power, flexible integrated memory, and high-speed serial I/O – that enable these devices to meet the high bandwidth requirements of video applications.

With these features, you can use Virtex-4 devices in a variety of products, such as cable modem termination systems, digital video broadcast systems, flat-panel displays, master control switches, MPEG encoders, non-linear video editors, broadcast routers, image statistical multiplexers, and video servers.

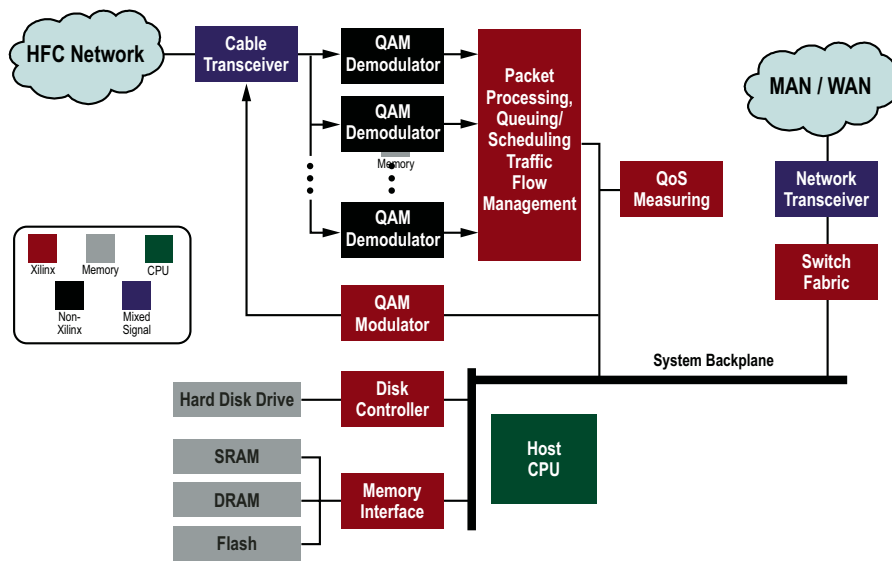


Figure 1 – Cable modem termination system block diagram

Cable Modem Termination System

One common application where you can use Virtex-4 devices is in a cable modem termination system (CMTS), shown in Figure 1. The CMTS is used in cable headends, a switching system that works in conjunction with Internet service providers to route data between cable modems and the Internet.

In a CMTS, the transmitted data is multiplexed onto a cable channel along with broadcast video transmissions. Bandwidth is shared by all active subscribers (typically 500 to 2,000) in the cable network segment. Downstream transmission rates run at 40 Mbps using quadrature amplitude modulation (QAM), while upstream rates can be as high as 10 Mbps using QAM or quadrature phase shift keying (QPSK). The speed of the upstream link depends on the service level agreement (SLA) that the subscriber has signed with their cable company.

CMTS Design Challenges

Cable operators can offer a variety of different services by using quality of service (QoS) provisioning to support different subscriber packages, helping to maximize their revenue stream. For QoS in the CMTS, the design needs to support packet classification, packet prioritization, flow control, congestion control, queuing, scheduling, and QoS statistical measure-

ments. All of these functions need to be supported without a reduction in user bandwidth. Given this, QoS processing is generally done in hardware, for software implementations lack the processing power to make real-time routing decisions and can result in delays and excessive queuing.

Maintaining efficient bandwidth utilization while supporting SLAs and multiple traffic types makes traffic management very challenging. Throw in varying protocols, memory management, different sized payloads, and a variety of different system interfaces, and it is easy to see how these designs require high-performance, cost-effective flexibility that ASSPs and ASICs cannot offer. These challenges open up opportunities for Virtex-4 devices that can provide flexible traffic management capability at the required performance levels.

CMTS Queuing and Scheduling Requirements

QoS provisioning is basically a queuing and scheduling problem. Proper queuing and scheduling entails recognizing service classes along with managing buffer memories and

port bandwidth. The design goal is to reduce the amount of congestion in order to offer the maximum amount of bandwidth and packet throughput by optimizing end-to-end delay and minimizing packet loss.

In addition, the implementation needs to support fair bandwidth distribution for each service class; furnish protection between the different class levels; provide fast, flexible access to bandwidth without impacting forwarding performance; and allow other service classes to use underutilized bandwidth.

To surmount these challenges, efficient queuing and scheduling techniques are required to optimize queue memory management, which controls the number of packets in a queue. This function controls service-class access to the packet memory buffer and determines which packets to drop because of congestion.

Multiple queue memory management techniques are in use today, including random early detection (RED), weighted random early detection (WRED) and leaky bucket. Per-flow queuing is commonly performed using one or a combination of the scheduling algorithms shown in Table 1.

Table 1 shows that there are many different queuing and scheduling algorithms. Given the dearth of standards activity in this area, many different algorithms will continue to exist for the foreseeable future. In addition, these algorithms need to handle variable sized packets, which

are more complicated than fixed cells.

Virtex-4 devices offer a high-performance solution for these queuing and scheduling requirements, for the devices offer an extremely fast and flexible fabric for implementing designs without impacting forwarding performance. Scheduling decisions are typically performed every clock cycle and require heavily pipelined designs.

Virtex-4 devices also offer a register-rich architecture with ample routing, enabling efficient implementation of these decisions. The high-speed designs also require very

Queuing and Scheduling Algorithms
First-In, First-Out
Round Robin
Weighted Round Robin
Fair Queuing
Weighted Fair Queuing
Priority Queuing
Shortest Remaining Time

Table 1 – Common queuing and scheduling algorithms

Additionally, the Virtex-4 memory-rich architecture, capable of running at 500 MHz, provides much needed on-chip cache capability.

wide internal buses, which are easily implemented in the Virtex-4 architecture by using the integrated DLLs and DCMs to help manage multiple clock domains.

Many of the queuing and scheduling buffer management schemes are math-intensive; these schemes must quickly calculate multi-variable equations such as packet transmit scheduling and customer service normalization schemes. For instance, the bandwidth calculation shown in Figure 2 is a multi-variable equation used to calculate the bandwidth (B1, B2) for each user for a given level of total band-

To properly interface to these memory devices, all Virtex-4 devices have the ChipSync™ feature in every device I/O. ChipSync lets designers easily align the DQS control signal with memory data in very small increments; this alignment can be easily monitored and altered as temperature and voltage changes alter the very delicate timing.

Converting the high-speed 300 MHz+ memory data to wider, slower, more manageable data is easily accomplished with the built in ISERDES and OSERDES available in every I/O. Additionally, the Virtex-4

The integrated distributed RAM is good for implementing small FIFOs, DSP coefficients, shallow/wide memories, and CAMs. The block RAM is good for larger FIFOs, packet buffers, video line buffers, cache tag memory, deep/wide memories, and CAMs. Xilinx also has many proven embedded-memory CAM and FIFO reference designs available to help implement these high-speed memory designs.

CMTS Video Transmission Standards

The ITU-T (International Telecommunications Union – Telecommunication Standardization Sector) has created a standard for the transmission of audio, video, and data services over cable networks. The specification for this standard is ITU-T J.83 Digital Multi-Program Systems for Television, Sound, and Data Services for Cable Distribution.

This standard is supported in Virtex-4 devices using the Xilinx J.83 Cable Modulator LogiCORE™ IP to provide either single- or quad-channel support. (See the related article from the Winter 2004 issue of the *Xcell Journal*, “Using System Generator for DSP to Create the J.83 Cable Modulator.”)

Conclusion

Given the high bandwidth requirements of a CMTS along with the associated queuing and scheduling complexities to provide the appropriate QoS requirements, Virtex-4 devices offer an optimal solution for these designs. The embedded hierarchy of memory structures, along with integrated high-speed serial interfaces and programmable flexibility, make Virtex-4 devices a better choice over implementations using ASICs or ASSPs.

To learn more about Xilinx key markets and end applications, visit www.xilinx.com/espl/. For more details on Virtex-4 FPGAs, visit www.xilinx.com/virtex4/.

Problem: Known the whole bandwidth (B), the Drop Probability (P1, P2, and P3), and the number of Flows for each PHG class calculate B1, B2, and B3

Total Bandwidth for the aggregate PHB AF1 group is: **B**

PHB **AF13**: [Drop Probability = **P3**, Bandwidth **B3**; Flows **N3**]

PHB **AF12**: [Drop Probability = **P2**, Bandwidth **B2**; Flows **N2**]

PHB **AF11**: [Drop Probability = **P1**, Bandwidth **B1**; Flows **N1**]

$$\begin{cases} N_3 P_2 B_2 = N_2 P_3 B_3 \\ N_3 P_1 B_1 = N_1 P_3 B_3 \\ B_1 + B_2 + B_3 = B \end{cases} \Rightarrow \begin{cases} B_i = \tilde{P}_i B \quad i = 1, 2, 3; \quad i \neq j, k \\ \tilde{P}_i = \frac{\hat{P}_j \hat{P}_k}{\hat{P}_i \hat{P}_j + \hat{P}_j \hat{P}_k + \hat{P}_i \hat{P}_k} \quad \hat{P}_x = \frac{P_x}{N_x} \end{cases}$$

Figure 2 – Bandwidth calculation formula example

width. These types of functions can take advantage of the integrated 500 MHz performance, low power, 18 x 18 multipliers, and 48-bit adder/subtractor integrated in the XtremeDSP™ slice.

CMTS Memory Requirements

Most networking applications are built around a load-store type of architecture, with packets being stored in linked lists in external memories. Because of the increasing queuing and scheduling performance requirements of the CMTS, high-speed DDR or QDR SRAM memories prevent memory access from becoming a bottleneck.

memory-rich architecture, capable of running at 500 MHz, provides much needed on-chip cache capability.

Virtex-4 devices support high-speed memory interfaces and, along with an embedded hierarchy of memory structures comprising distributed and block RAM, can easily facilitate implementation of high-performance queuing and scheduling algorithms. The Virtex-4 devices' high memory-to-logic ratio helps reduce memory access latency by caching data on-chip, buffering data between two disparate clock domains, and using scratch-pad memory for storing coefficients.

We Have What You've Been Looking For



Flexible Design

Powerful Performance

Greater Programmability

Advanced Technology

- The **Memec Virtex-4 Development Kits** are the ideal solution for designers needing a high-performance Virtex-4 platform with the flexibility to meet your system design challenges.

Your Search is Over. 



Visit www.memec.com/xilinx-v4



Developing Next-Generation Telecommunication Networks

Virtex-4 FPGAs provide the density, features, and performance at low price points to enable the communication revolution.

by Amit Dhir
Senior Manager, Strategic Solutions,
Wired Networks and Telecom Markets
Xilinx, Inc.
amit.dhir@xilinx.com

Although the dot-com bubble may have burst, the Internet has continued its multi-fold growth, thus placing a strain on telecommunication networks. Both individuals and businesses are demanding more bandwidth to run new communications options, such as desktop video conferencing, IP telephony, remote storage, and mobile communications.

This is the driving force behind the need to transform the multiple, costly, and complex networks in use today into a smarter, multipurpose, global, cost-effective broadband network. This transformation will generate new sources of revenue for service providers, provide greater opportunities and productiveness for enterprises, and meet the needs of consumers who value multimedia, the freedom of mobility, and personalized and secure private network services. The boundaries between public and private, wired and wireless, and voice and data networks are vanishing.

The key elements of a more intelligent, high-speed, multi-purpose global network include broadband and optical technologies, voice over packet, wireless data, multimedia services and applications, and security, all underpinned by a packet network core. Typical telecom- and datacom-wired equipment can be segmented into line cards, switch cards, control cards, and a backplane. Network convergence requires equipment vendors to support multiple technologies, including SONET/SDH, PDH, Data over SONET (GFP, VCAT, and LCAS), Fibre Channel, Ethernet, DVI, DSL, PON, and MPLS, depending on the system's location in the access networks, metropolitan area networks, enterprises, and wireless networks.

Because data is transmitted in IP packets, packet processing has become a sophisticated architectural decision depending on the end system. This also influences the switch architecture and backplane topology. Also, with time to market and cost pressures, equipment providers continue to focus on technology and innovation as the cornerstones for creating new revenue opportunities.

Enabling the Communications Revolution

Xilinx® FPGAs offer a high-performance fabric, integrated features, and powerful clock management, thus providing an ideal platform for communications equipment vendors to develop their solutions. Xilinx also provides case studies, IP, and reference designs to help customers with their designs in several key applications.

Telecom and Datacom Line Card Port Interfaces

Digital telecom infrastructure has mostly been based on PDH and SONET/SDH technologies in the metropolitan area and transport networks. The transport of data traffic (Ethernet, Fibre Channel, ESCON, and DVI) onto SONET/SDH networks is giving rise to technologies such as generic framing procedure and virtual concatenation. This flux is requiring a need for pro-

grammable solutions that can allow vendors to have a single SFP or XFP module to support multiple technologies at given rates. With the Virtex-4™ FX family supporting Gigabit Ethernet (1 and 10 Gbps), Fibre Channel (1, 2, 4, 8 and 10 Gbps), and SONET (OC-12 and OC-48) on every RocketIO™ serial transceiver, you have extreme flexibility in the I/Os.

The FPGA, coupled with robust IP offerings from Xilinx and our partners for MACs and framers/mappers, presents a flexible solution that can be morphed

already enabled several customers to upgrade their backplane to faster rates.

With the Virtex-4 family's third-generation multi-gigabit transceivers and enhanced features such as AC coupling, programmable preemphasis, and receive (linear and decision feedback) equalization, you can ensure signal integrity in a wide variety of applications and give new life to old systems by upgrading legacy backplanes.

Industry standards such as Serial RapidIO™, Gigabit Ethernet, and PCI Express (including out-of-band signaling

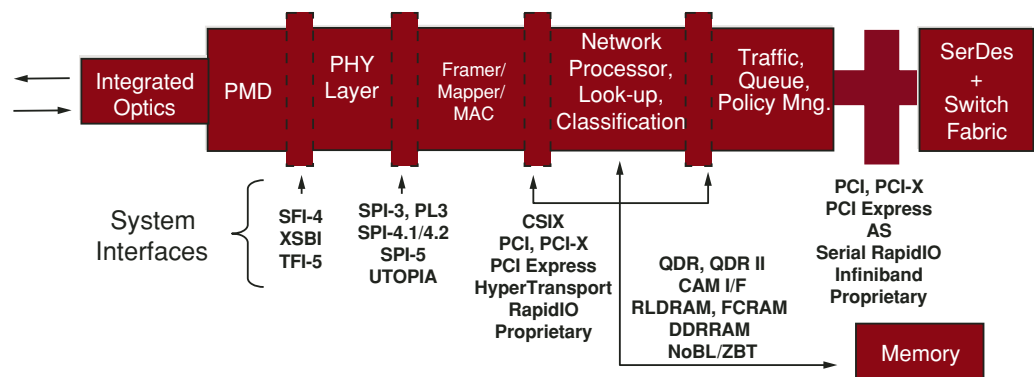


Figure 1 - Typical line card

depending on the service provider's needs on a per-port basis. This also helps in the lifecycle cost management of the system, as fewer cards need to be maintained and can be programmed with the relevant port interfaces required upon shipping.

Serial Backplanes and Switching

With exploding data rates and source synchronous I/Os unable to keep up with the pace at which packet communication occurs between the line cards, vendors are universally looking at serial technologies to solve the bandwidth problem. RocketIO transceivers, which support a wide performance range of 622 Mbps to 11.1 Gbps, can also be used to drive several tens of inches on FR-4 and other exotic materials – at different rates. With Virtex-II Pro™ and Virtex-II Pro-X families and the integrated RocketIO transceivers, Xilinx has

and spread spectrum clocking) are all supported. Virtex-4 FX FPGAs enable bridging between just about any serial or parallel system interface.

To enable the creation of mesh designs, Xilinx offers the mesh fabric reference design for complete flexible connectivity across a serial backplane based on the standard of your choice. Xilinx also provides signal integrity tools and resources such as the ATCA development board to ease the process of designing SerDes solutions into your next-generation backplane.

Packet Processing

Although several network processor vendors have attempted to solve packet processing (classification, policing, queuing, and scheduling) glitches, achieving performance and power goals continues to be challenging. Virtex-4 FPGAs solve network process-

ing challenges with features such as system and memory interfaces, clock managers, block RAM, DSP slices, PowerPC™, and high-speed programmable logic. Xilinx also offers solutions such as the queue manager and mesh fabric reference designs to help with traffic management needs.

Simplifying System Design Challenges

The fundamentals of unparalleled flexibility and high performance are further extended in the Virtex-4 family. To help simplify your system design challenges, Xilinx also offers:

- **Integration.** The integration of processors, tri-mode Ethernet MACs, DSP slices, SerDes, memory, and other features in the FPGA helps reduce your bill of materials and saves FPGA resources. This reduction in component count helps streamline logistics with a smaller bill of materials and simplifies the design and manufacture of system hardware because of simpler PCB design and manufacturing and improved reliability through the reduction of solder joints.
- **SelectIO™ technology and connectivity IP.** Virtex-4 FPGAs make it easy to build robust high-speed memory and networking interfaces. All Virtex-4 platforms include configurable, high-performance SelectIO technology to support a wide variety of I/O standards.

Virtex-4 FPGAs provide as many as 960 user I/Os, supporting more than 20 single-ended and differential electrical I/O standards to enable several parallel system interface standards on one device. New ChipSync™ technology built into every I/O block makes source-synchronous interfacing to the latest high-speed components easy. Plus, powered with XCITE technology, each I/O block delivers on-chip active I/O termination, eliminating external termination resistors to increase signal integrity, save board space, and reduce system cost. Xilinx also provides a robust offering of IP (PCI, SPI-3, SPI-4.2, RapidIO)

and reference designs (DDR2, DDR, QDR II, RLDRAM II, FCRAM II) for system and memory connectivity.

- **Embedded processing.** With the embedded PowerPC and the soft MicroBlaze™ and PicoBlaze™ processors, Xilinx offers a range of processing solutions to match the requirements of different tasks, ranging from simple control functions to advanced algorithms and high-speed calculations. Also, in telecom cards the processors assist with simple functions such as alarm handling and performance monitoring.
- **Low-cost designs.** Xilinx manufactures Virtex-4 FPGAs using 90 nm advanced process technology on 300 mm wafers. This allows us to produce approxi-

mately five times as many die per wafer, compared to building an equivalent chip in 130 nm process on 200 mm wafers. This lowers the cost per die significantly.

Additionally, the EasyPath™ program further lowers system cost for customers who are ready to take their finished design to volume production. Xilinx creates customized test programs for EasyPath customers that exercise only the device resources used in the specific design. This approach shortens test time and increases yield to reduce FPGA unit prices as much as 80%.

Conclusion

To learn more about the key markets and end applications of Xilinx solutions, visit www.xilinx.com/espl/ or e-mail espteam@xilinx.com. For more details on Virtex-4 FPGAs, visit www.xilinx.com/virtex4/. 

FREE Online Seminar Verification of Your Embedded FPGA Design

Seamless FPGA for Xilinx Virtex-II™ Pro



During this session, you will learn how to:

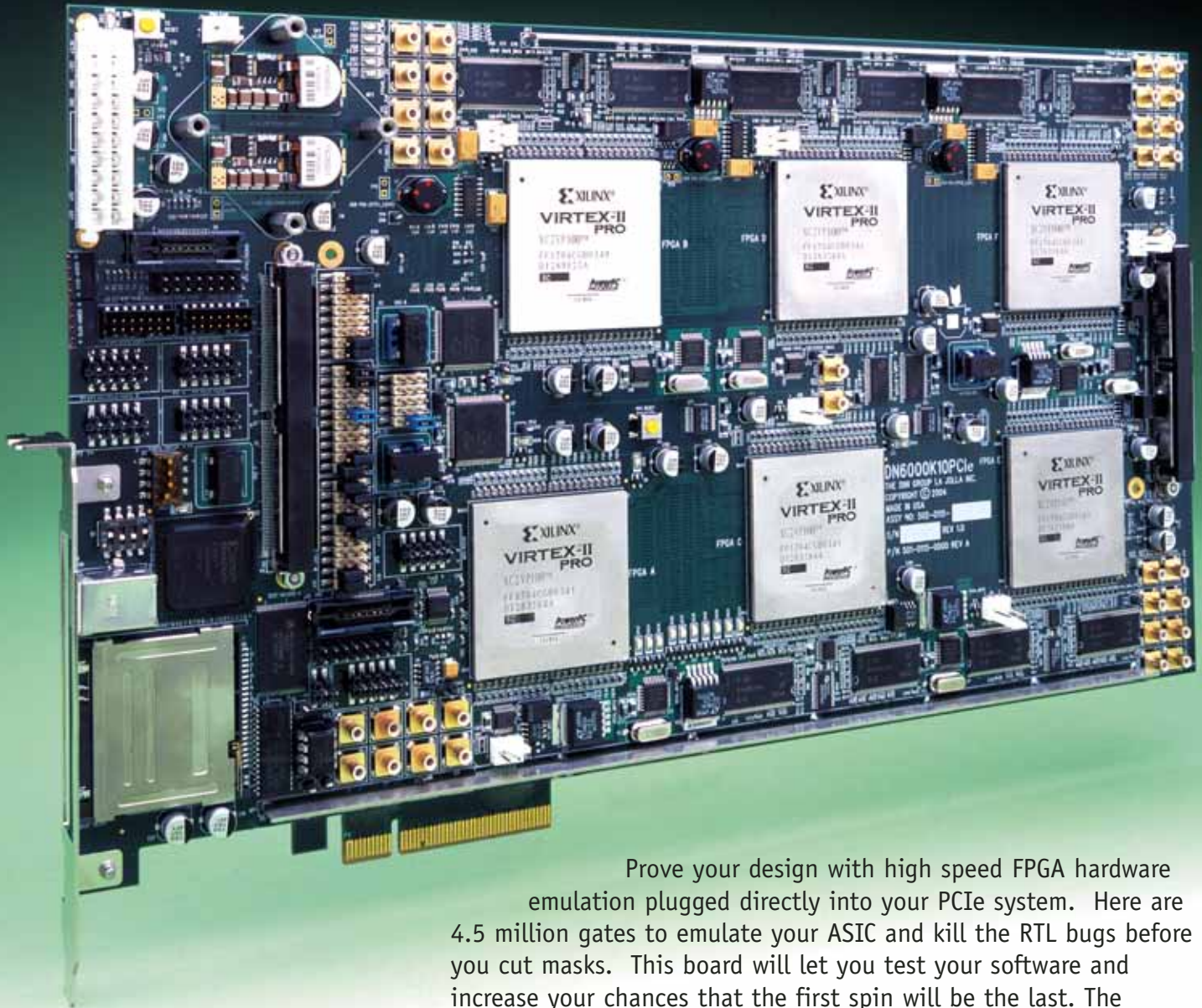
- Leverage Platform FPGAs for embedded systems
- Utilize the tightly integrated solution of Seamless with Xilinx Platform Studio (XPS)
- Easily debug complex hardware-software interactions
- Measure software and hardware performance of the FPGA system

Learn more today: <http://www.mentor.com/fv/events/seminars/xilinxonline/>

For additional details about Seamless FPGA, visit us at www.seamlessfpga.com



Prototype Your PCIe ASIC *HERE*



Prove your design with high speed FPGA hardware emulation plugged directly into your PCIe system. Here are 4.5 million gates to emulate your ASIC and kill the RTL bugs before you cut masks. This board will let you test your software and increase your chances that the first spin will be the last. The DN6000K10PCIe is packed with the features you need:

- 1,4 and 8-lane versions
- Six VirtexII-Pro FPGAs (-2vp100s, the big ones)
- 10 DDR (64Mx16) and 4 SSRAMs (2Mx36) external to the FPGAs
- Expansion capability to customize your application
- Synplicity Certify® models for quick and easy partitioning

Like all our products, this new PCI Express bus board will help you get your ASIC to market on time and in budget. Call The Dini Group today-- PCIe is already here.

The
Dini
Group

Virtex-4 FPGAs for Software Defined Radio

Red River's new PCMCIA Type II module can transform any notebook computer into a software defined radio using a Virtex-4 FPGA for performance-critical DSP functions.



by Ken Sienski
President
Red River
sienski@red-river.com

Established in 1996, Red River specializes in high-performance signal processing and data communication solutions for the embedded systems market, especially software defined radio applications.

Our main challenge in serving the software defined radio market is to have a hardware platform that meets the demands of multiple configurations. Some customers are looking for a complete, pre-built radio solution; others are looking to add custom features to a radio platform. These disparate requirements place great demands on us to find a common programmable silicon solution that meets both needs.

The Xilinx® Virtex-4™ FPGA family

allows us to do exactly that – provide different customer solutions at the lowest cost. Advanced features such as FIFO logic, embedded PowerPC™, RocketIO™ transceivers, and Ethernet MAC, as well as advanced power and packaging technology, makes Virtex-4 devices a perfect choice for us.

Model 351 (Pocket Change)

Our next-generation product, the Model 351, or “Pocket Change,” transforms any portable computer into a high-performance multi-channel software defined radio transceiver. The Pocket Change CardBus PC Card accepts two analog input signals through MMCX coaxial connectors on the outside edge of the card. The receiver input is AC-coupled to a 14-bit (80 MSPS) A/D converter. The transmitter output is supplied through a 14-bit (100 MSPS) D/A converter. Most of the

digital logic is supplied using a Virtex-4 FPGA device.

When we began developing the Model 351, we investigated various offerings on the market and finally decided to use Virtex-4 FPGAs. The Virtex-4 FPGA family provides the flexibility and features that support both our needs and the requirements of our customers.

The Model 351 design comprises a Virtex-4 FPGA connected to an A/D converter, a D/A converter, and a dedicated PCI bus controller (for the CardBus interface to the host computer) (Figure 1). Although it is targeted at our traditional software defined radio customers, the Model 351 is also suitable for signal acquisition or generation, signal intelligence collection, transceiver modem algorithm prototyping, frequency hop signal generation, or portable signal recorder/playback applications.

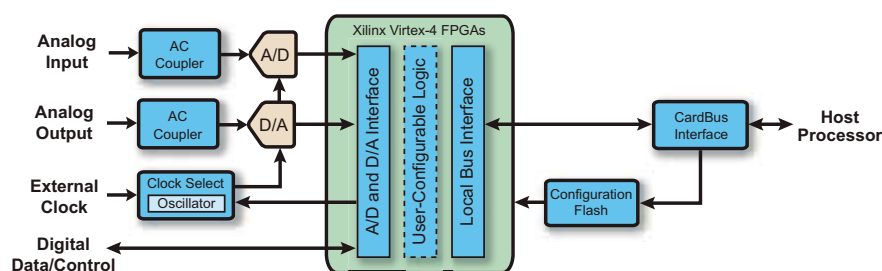


Figure 1 – Model 351 block diagram

Customization and Flexibility

Initially we considered using dedicated digital upconverter/downconverter chips to implement the Model 351 transceiver function. However, many of our customers prefer the flexibility of inserting custom functions into their designs. The customization requirement pushed us to use programmable technology.

By selecting a leading programmable logic architecture, we can address the customization needs of a broad set of customers. Xilinx ISE™ development software provides our customers a familiar design environment to embed custom DSP functions in the uncommitted logic of the Virtex-4 FPGA.

Another benefit from using Virtex-4 FPGAs is that we can offer multiple products using one common hardware platform. This has helped reduce hardware development time and simplify inventory management.

Power and Space Efficiency

One of the challenges in CardBus PC Card development is to select a device that meets the PCMCIA functional specification and the tight power restriction of 3.3W. We were impressed with the power efficiency of the Virtex-4 family, as it consumes half the power of comparable logic solutions.

Virtex-4 FPGAs give us significant features and performance while still meeting the tight power budget of our design. In addition, PCMCIA imposes severe height restrictions in order to fit into the Type II module form factor. The Virtex-4 FF668 package offering is one of the few FPGA packages that meet the height requirements.

Advanced Features and Performance

One key requirement for a software defined radio application is high-performance DSP capability. The performance requirement is driven by the need to support multiple signal channels in real time.

Virtex-4 FPGAs are capable of performing multi-channel digital upconversion and downconversion across the entire Model 351 analog bandwidth. The Virtex-4 device can also perform Fast Fourier Transforms (FFTs) for spectral analysis of incoming signal data.

The Virtex-4 FPGA provides the “heavy lifting” to process digital information between the host computer and the A/D or D/A converter. The signal processing power comes directly from the SX platform. Virtex-4 devices can achieve high-DSP performance by taking advantage of massive parallelism within each FPGA. For math-intensive algorithms (like DUC/DDC applications in a software defined radio), the high number of DSP slices – multiply/add/accumulate engines – that can run up to 500 MHz provides the kind of performance only previously available in fixed ASIC technology.

Our designs also make extensive use of the internal block memories in the FPGA to provide multi-queue FIFO capabilities. The FIFOs are used to buffer data between the A/D or D/A converters and the local bus for DMA operations, providing performance-intensive processing without involving the host CPU in memory transfers. This gives our products the ability to flexibly handle digital radio data without completely consuming the CPU performance of the host computer. With

the highest-performance internal block RAM and unique integrated FIFO logic, Virtex-4 FPGAs give us the FIFO quantity and performance that we need to keep up with the bandwidth of the analog components and host interface.

Three Platforms Satisfy Multiple Requirements

The three Virtex-4 platforms (LX, SX, and FX) give us unique capabilities for several upcoming products. For customers wanting to add custom logic functionality, we use the LX platform. LX offers the choice of many different gate densities within the same package footprint, allowing us to use the same base design to support many different customer needs.

We have some designs that necessitate tremendous additional DSP capability for math-intensive processing, including signal modulation and demodulation. For these applications, we see the SX platform as a natural fit. SX devices give us by far the largest amount of DSP performance.

For some of our other designs, we are implementing the advanced system-level block functionality of the FX platform – PowerPC running VxWorks, RocketIO transceivers for optical and PCI Express interfacing, and gigabit Ethernet MAC cores. Because Virtex-4 devices give us three platforms to choose from, we can offer different capabilities across our product line.

Conclusion

Software defined radio products must address a broad application space, which presents a challenge when selecting component features. The three Virtex-4 platforms give us the feature choice and performance that we require to field a family of solutions for both fixed and mobile installations.

The upcoming Model 351 demonstrates cutting-edge capabilities in an extremely small, power-efficient module that operates in a standard notebook computer. Visit www.red-river.com for more information about the Model 351 and other Red River products. 

Virtex-4 FPGAs in Rugged LCD Monitors

Integrated features like ChipSync technology not only reduce cost but improve ease of use and design cycle time.

by Fabrice Mommens
Project Manager, Defense & Security Lab
BarcoView Command & Control
fabrice.mommens@barco.com

Using in-depth market knowledge, Barco designs and develops solutions for large-screen visualization, display solutions for life-critical applications, and systems for visual inspection. Barco is currently active in the traffic, surveillance, broadcasting, presentation, simulation and virtual reality, edutainment, events, media, digital cinema, air traffic control, defense and security, medical imaging, avionics, and textile industries.

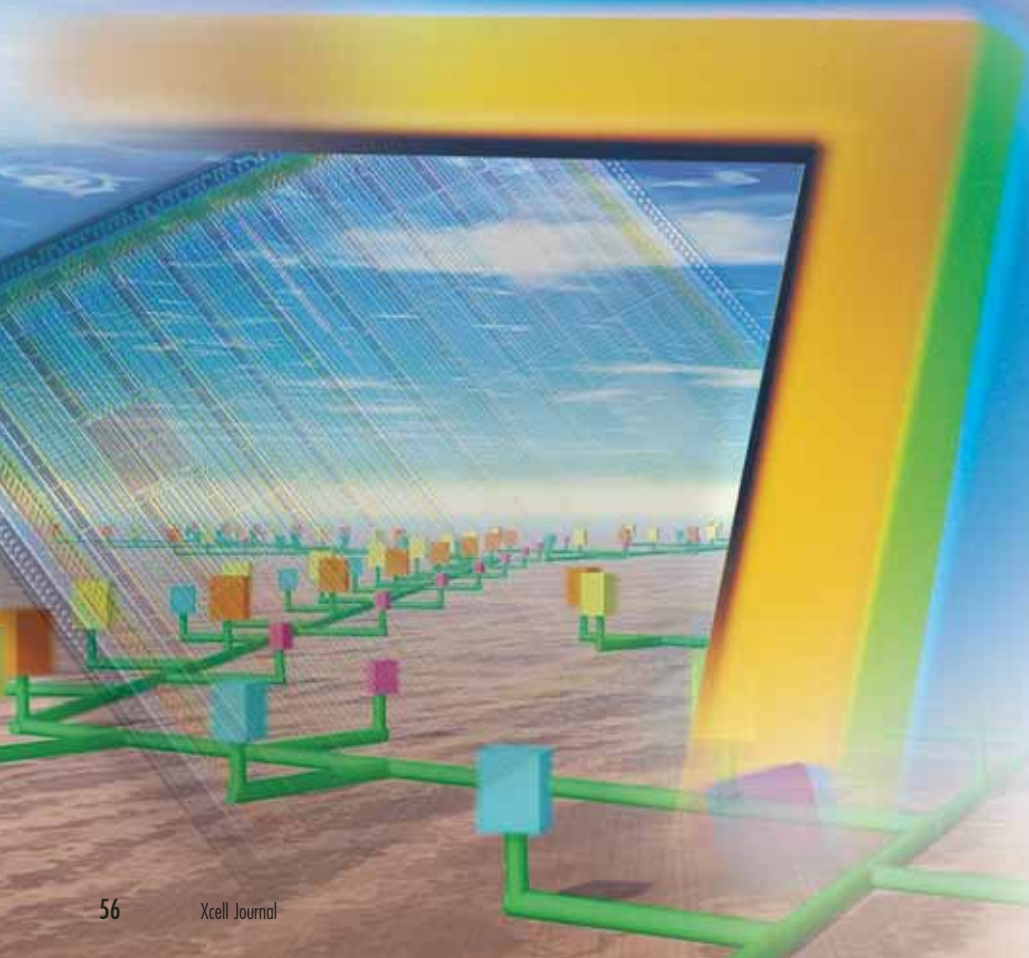
My particular division at Barco, BarcoView Command & Control in Belgium, has been a Xilinx® customer for just over two years. Our division's choice to standardize on Virtex™ products was based on the availability of the embedded PowerPC™ processor, first introduced by Xilinx in their Virtex-II Pro™ product family.

We like to design with FPGAs in our systems because they can be reprogrammed throughout the life of the product. This critical feature allows us to add features from one generation to the next without having to redesign the whole system.

BarcoView Command & Control is working on a rugged family of LCD monitors. These products are designed for rough environments where commercial display products would not survive. In these designs, FPGAs are mainly tasked to perform video and image processing.

The system is currently designed around a Virtex-II Pro device, in which the PowerPC processor, running a real-time embedded operating system, controls the complete display system. Looking at the new features of the Virtex-4™ FX family, we are planning to migrate these Virtex-II Pro designs that use the PowerPC processor to Virtex-4 FX devices in a future version of the project.

Besides the central control of the display system, we also use FPGAs in the data path for specific processing. The part of the design where we chose to implement the Virtex-4 FPGA is an optional feature of the displays, where it performs real-time image scaling on the video stream.



This scaler module can receive a video stream on its input at a very high rate (160 MHz x 24 bits = 3.84 Gbps), perform scaling on the stream, and send out the scaled stream at the same rate. With the amount of data being processed and because of the way the scaler algorithm works, we must store the incoming video stream into memory before processing it. Thus, we had to look at very fast external memories (DDR2).

video stream is written back into an output memory buffer at 100 MHz on 24 bits. The output memory buffer can then be read at a frequency reaching 160 MHz on 24 bits to further process the data. After all that processing and some more, the images are displayed on the LCD monitor.

As shown in Figure 1, which represents the Virtex-4 LX15 ecosystem of our design, the memory bandwidth requirements for the

ChipSync technology allows us to easily reach 400 Mbps and intuitively design this interface. Without this feature, we would have needed a 32-bit interface to the external memory. Though running at half the clock rate, more physical SDRAM on the board would be required, as there is no such thing as a small SDRAM device. In addition to the higher unused memory locations, we would have required a larger package for the scaler device because of the increased number of pins, using more board real estate.

ChipSync technology also allows us to easily use DDR-2 interfaces, enabling us to choose the very latest in SDRAM technology. This helps to avoid obsolescence issues, a common problem in the memory industry.

Block RAM: Not Just Memory

Another critical point when choosing the right FPGA is the amount of block RAM available in the device. Having flexible, fast internal RAM is a critical factor for us because we use block RAM for two things: as video line memory and as FIFOs for the DDR-2 memory controller. Smaller, slower, or less flexible RAM blocks would have produced a more complex DDR-2 memory controller design, resulting in larger logic requirements and therefore a larger device.

In addition to speed, flexibility, and size, the integrated FIFO logic available on each block RAM allows us to save a substantial amount of logic and guarantees fast FIFO operation, simplifying the design of our whole system.

Conclusion

The logic savings obtained through the use of the integrated FIFO, ChipSync technology, and the use of smaller external memories results in a significant cost reduction. Additionally, the ease of use, implementation, and modification brought by the hard IP blocks makes the Virtex-4 LX15 device perfect for this application.

After designing with the Virtex-4 LX FPGA, we are looking forward to evaluating the Virtex-4 FX platform to see how we can benefit from all the new features available with the integrated PowerPC processor.

For more information about Barco and our products, visit www.barco.com.

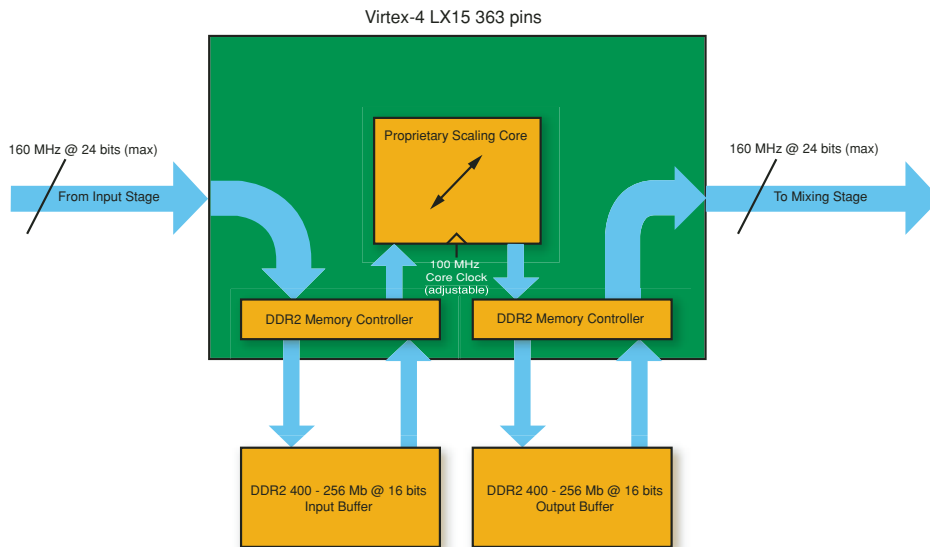


Figure 1 – Video scaler block diagram based on a Virtex-4 FPGA

Memory Interfaces Made Easy

When searching for the right product for our application, we looked at many alternatives. However, it rapidly became clear that Virtex-4 devices could best perform the required tasks.

The main reason for choosing Virtex-4 FPGAs was the availability of the ChipSync™ feature, with support for DDR-2 400 memories. Having support for DDR-2 400 gives us enough bandwidth to reduce the number of physical RAM chips needed, reduce the board real estate, and in the end reduce system cost.

Looking at the data flow, these video streams are digitized into pixels up to 24-bit RGB (it could be a narrower stream depending on the input source). The incoming stream is stored into an input memory buffer at a frequency reaching up to 160 MHz. The data from this input memory buffer is then fed to the scaler core, also on 24 bits, at a maximum frequency of 100 MHz.

After the core has processed the data, the

input and output buffers are identical. Focusing on the input memory stream, we can see that the bandwidth required is $(160 \text{ MHz} + 100 \text{ MHz}) \times 24 \text{ bits} = 6,240 \text{ Mbps}$.

This is where the advantages of 400 Mbps DDR-2 are realized. Because of this memory speed, we can select a 16-bit-wide DDR-2 SDRAM running at 200 MHz and still have enough bandwidth to process the input memory buffer streams (the stream coming from the input source and the stream going to the scaler core).

A simple calculation shows that $200 \text{ MHz} \times 2 \text{ (double data rate)} \times 16 \text{ bit} = 6,400 \text{ Mbps}$. This is higher than the 6,240 Mbps previously calculated for the input buffer. Of course, we need to take into account a small overhead for the memory controller (during transients), but the margin should be more than enough to guarantee reliable system operation. If for any reason the controller's overhead becomes such that we cannot guarantee that the system would work properly, we can always lower the 100 MHz core frequency.

ISE 6.3i Software — Unleash the Power of Virtex-4 FPGAs

New ISE technology delivers breakthrough performance with greater ease of use.

by Lee Hansen
Sr. Product Marketing Manager
Xilinx, Inc.
lee.hansen@xilinx.com

The advanced silicon features introduced with Xilinx® Virtex-4™ FPGAs are readily available through ISE™ (Integrated Software Environment) 6.3i technology. This latest release of Xilinx design software comes ready to deliver maximum design performance, with new features and optional tools that will speed your Virtex-4 project to completion.

Advanced Timing Closure and Performance

ISE software lets you get the most out of Virtex-4 devices and your target project. Benchmark testing on a suite of real-world, customer-based designs demonstrates that Virtex-4 FPGAs, with ISE 6.3i design software, are as much as 43% faster than the nearest competitive FPGA. On average, that's an extra speed grade advantage.

The performance-driven ISE technology — like our exclusive timing-driven map option — helps you achieve better design packing and better performance, particularly if your target device is already more than 90% utilized. Timing-driven map can yield 30% better overall design performance depending on design utilization.

This additional performance advantage gives you the potential to stay in a lower density target Virtex-4 device, even if utilization is pushing 90% or higher, when competing tools would have already forced the design into a larger, more expensive device.



High-Density Design

ISE design software also includes a full spectrum of tools for larger density designs, including area and logic group floorplanning, incremental design for faster design recompile cycles, and modular design for team-based project approaches. High-density designers can also separately purchase the new PlanAhead™ hierarchical floorplanner, which wraps all of these methodologies into one separate advanced tool. Together, these tools augment the design flow of high-density projects with methodologies that speed through to project completion, as well as performance-locking strategies to help bring large designs under control.

Area Groups

Using either PACE (Pinout and Area Constraints Editor) or ISE Floorplanner, both included with all configurations of ISE design software, you can quickly floorplan areas of logic from your design onto your target Virtex-4 device. You can create area groups around hierarchical HDL boundaries, or let PACE create default area estimates for target logic, or draw logic areas by hand.

Visualizing the different areas of logic helps you partition out areas for design reuse or IP placement, or section off where the “tough” areas of the design will be concentrated. Most importantly, area planning can help accelerate timing closure by grouping critical logic and paths together, and minimize the number of interface points between modules.

Modular Design

ISE design software also includes modular design, a capability that implements a “divide and conquer” strategy for large designs – and for the corporate environments that deploy teams of engineers to tackle them. A design team manager first plans the design project, using floorplanning to partition the overall larger project into smaller design “modules.” These modules can then be assigned to individual team members for completion independent of the other modules.

Completion is focused on only that particular module of the overall design, with all teams completing their work in parallel.

Once a module is finished, its place and route results are locked while the project manager waits for the remaining modules to be completed.

Modular design delivers full planning control over the larger design, implementing a true bottoms-up design approach that completes the larger project much faster.

Incremental Design

Incremental design, also included with ISE design software, combines the quick-and-easy facet of area groups with the performance-locking aspects of modular design to deliver faster runtimes during heavy design iteration cycles.

Using PACE, you can assign area groups along hierarchical HDL boundaries; the overall design is then completed as usual. Should an incremental change become necessary, incremental design guarantees that you only have to re-implement the logic area that needs to change. The remainder of the design stays locked and intact, drastically speeding up overall compile times.

Incremental design also lets you make full use of the verification phase by delivering much faster overall project compile times. You can tweak critical design areas or implement ECO design changes late in the cycle with minimal impact on the larger FPGA project.

PlanAhead

In June 2004, Xilinx announced the acquisition of the leading-edge PlanAhead hierarchical floorplanner, developed originally by Hier Design. The PlanAhead floorplanner is a separately purchased tool option to the ISE design flow that is ideal for Virtex-4 high-density designs.

The PlanAhead tool utilizes an ASIC-style floorplanning methodology using a block-based approach. It enables you to analyze, detect, and correct potential implementation problems earlier in the design cycle, leading to the following benefits:

- Quicker incremental design changes
- Faster place and route
- Greater consistency and predictability in place and route
- Fewer design iterations
- Improved design performance
- Tighter utilization control
- Reuse of intellectual property and teamwork

The majority of low-density FPGA designs are implemented flat, with no hierarchy. Standard PLD place and route algorithms use more compile time to complete a flat design. By breaking the designs into

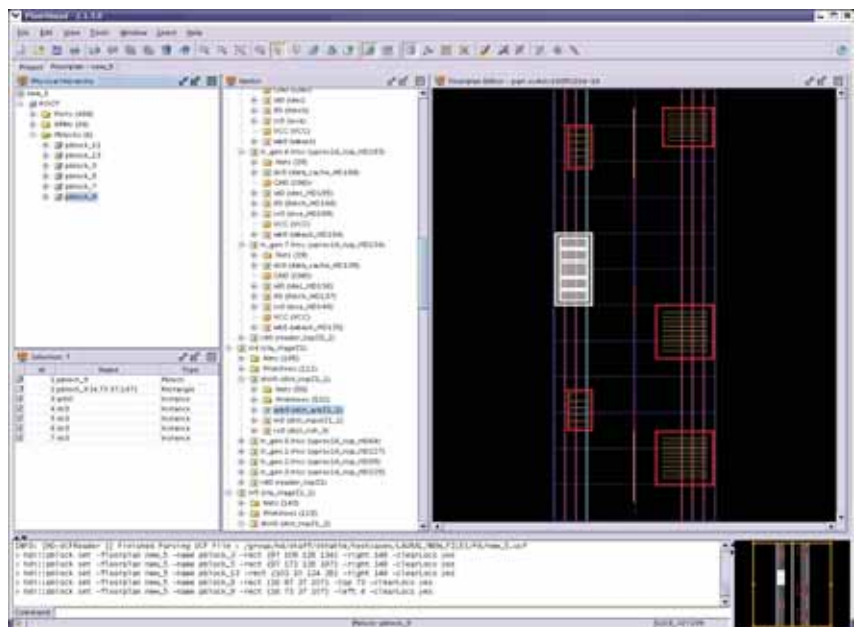


Figure 1 – PlanAhead floorplanning with a Virtex-4 LX100 FPGA

smaller pieces, or blocks, place and route doesn't need to converge on the entire design timing each time an incremental design change occurs. Hierarchy allows you to take maximum advantage to reduce place and route time.

You can also lock placement results for individual blocks that already meet timing so that subsequent place and route iterations do not change their performance, further stabilizing the overall design and making the overall results more consistently predictable. The PlanAhead tool wraps area groups, incremental design, and modular design into a single ASIC-strength floorplanner. Figure 1 shows Virtex-4 floorplanning using the PlanAhead hierarchical floorplanner.

Speed the Design Flow – ISE Architecture Wizards

The architecture wizards are a series of menus and dialog boxes built into all ISE configurations. These graphical menus let you quickly set advanced configuration parameters for FPGA silicon features. The wizards then write out editable VHDL or Verilog™ source code that is instantiated directly into your target project.

For example, the clocking wizard lets you easily set clock frequency, phase, multiplier factors, and delay for Virtex-4 devices and other Xilinx FPGAs using DCMs (digital clock managers). With the architecture wizards, you can rapidly set up and program advanced FPGA features, so even novice users can learn the most advanced Virtex-4 capabilities quickly.

Also new in ISE 6.3i software are two Virtex-4-exclusive architecture wizards, the ChipSync™ and XtremeDSP™ slice wizards. The ChipSync wizard configures groups of I/O blocks into an interface for use in memory, networking, or other types of bus interface design. You can quickly define key parameters such as the width and I/O standard of the data, address, clocks/strobes, clock buffers, and data bus specifications. All information is then presented in a clear and concise table for review.

The XtremeDSP slice wizard, shown in Figure 2, provides easy control of the revolu-

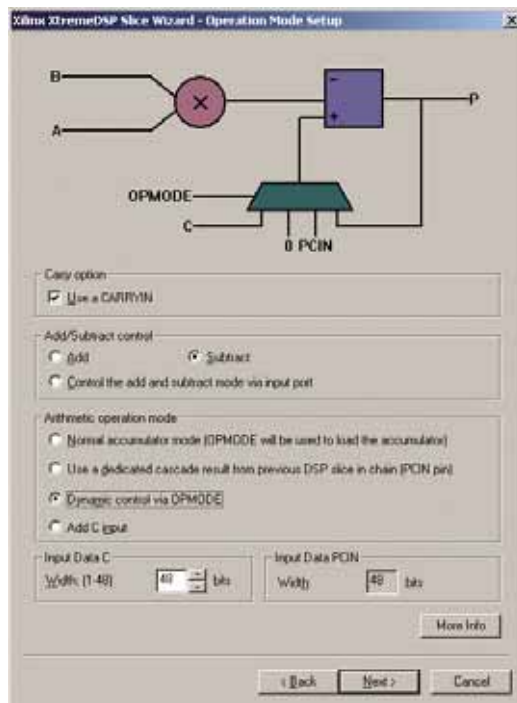


Figure 2 – XtremeDSP slice architecture wizard

tionary Virtex-4 XtremeDSP slice technology. This new silicon capability lets you build high-performance DSP filters and custom pre or post-co-processing DSP algorithms. The XtremeDSP slice wizard lets you specify accumulator, adder/subtractor, multiplier, or multiplier and adder/accumulator DSP modes. You can graphically set input and output bus data widths, pipelining options, clock enable, and reset pin setups, and then review parameters and output the results as HDL-ready code.

50% Faster Verification Cycles

Verification is one of the most time-consuming and time-critical phases of the design flow. As with most logic design suites, HDL verification and timing analysis are available. The ISE tools also link to additional verification technologies unique in FPGA design, including formal equivalence verification through Formality from Synopsys™ and Prover eCheck from Prover Technology AB, making quick work of verifying Virtex-4 high-density designs.

The ISE design tools also link directly to our optional, separately purchased ChipScope Pro™ real-time debug environ-

ment. ChipScope Pro tools insert low-profile logic analyzer, bus analyzer, and virtual I/O software cores during design capture. These cores are then synthesized and implemented into your silicon, allowing you to view:

- Any internal signal within the FPGA
- Embedded processor signals, including the IBM™ CoreConnect processor local bus or on-chip peripheral bus supporting the PowerPC™ 405 inside Virtex-4 FX family devices
- Embedded processor signals for the MicroBlaze™ soft-processor core

Signals are captured at or near operating system speed and brought out through the programming interface, freeing up pins for your design, not debug. You can then analyze captured signals through the ChipScope Pro software logic analyzer.

The ChipScope Pro environment also links internal FPGA debug to Agilent Technologies™ bench-top logic analyzers using the included ChipScope Pro ATC2 core. This core synchronizes the ChipScope Pro tool with Agilent's FPGA Dynamic Probe software.

This unique partnership between Xilinx and Agilent delivers deeper trace memory, faster clock speeds, and more trigger options, all using fewer pins on the FPGA, making Virtex-4 design debug as much as 50% faster than other logic verification methodologies.

Conclusion

You can unlock the power of Virtex-4 FPGAs with the ISE 6.3i FPGA environment, the most complete available for programmable systems design. Whether your design includes DSP, embedded, and high-speed serial I/O design, Xilinx ISE software and our optional System Generator for DSP, ChipScope Pro, and EDK and Platform Studio products will get your Virtex-4 LX, SX, and FX designs running with the maximum performance, while shortening design cycles and getting you to market faster. 



FIFOs Made Easy

Virtex-4 FPGAs have a complete FIFO controller in each block RAM.

by Peter Alfke
Director of Applications Engineering
Xilinx, Inc.
peter.alfke@xilinx.com

A FIFO is a memory subsystem where a data sequence can be written and retrieved in exactly the same order. No explicit addressing is required, and the write and read operations can be completely independent, using unrelated clocks.

"First-In First-Out" has been used in accounting for hundreds of years, as well as in data queues since the early days of computers. In 1970, Fairchild Semiconductor introduced the first integrated FIFO, the 3341.

Today, dedicated and much larger FIFO ICs are available, and mid-sized FIFOs are often implemented in Xilinx® FPGAs using the dual-ported block RAMs supported by soft cores for addressing and control.

A FIFO is an ideal subsystem: simple and user-friendly on the outside but complex and demanding in its implementation details. The design seems to be trivial; using a RAM with two independently clocked ports (one for writing, one for reading) plus two independent address counters to steer write and read data.

It may look easy, but the difficulty is found when you look deeper into the challenge – specifically, the decoding and synchronization of the obligatory status outputs indicating the extreme conditions of EMPTY and FULL. Even experienced

designers have had problems decoding these two conditions in a fail-safe way, especially when the FIFO operates with two independent clocks of several hundred megahertz.

Because fast asynchronous design is notoriously difficult, Virtex-4™ FPGAs now have a dedicated FIFO addressing and control circuit right inside each block RAM. Using the Virtex-4 block RAM FIFO option, you can be assured of reliable operation at a clock rate up to 500 MHz, without using any logic slices in the Virtex-4 fabric.

Virtex-4 FIFO

The FIFO shown in Figure 1 behaves like a "black box." You supply the data (4, 9, 18, or 36 bits wide), a continuously running write clock and its enable signal, and a continuously running read clock and read clock enable. Output data has the same width as the input data, unlike the basic block RAM where the two widths can be different.

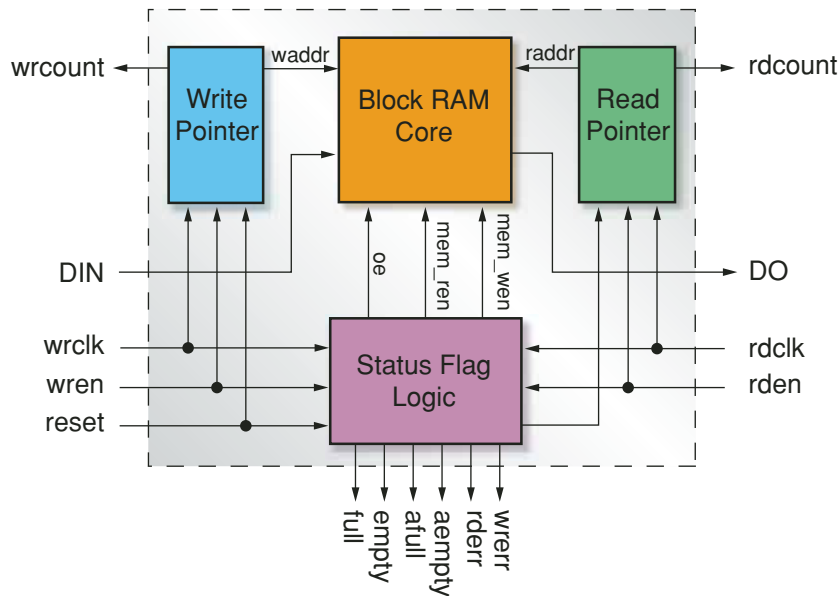


Figure 1 – FIFO block diagram

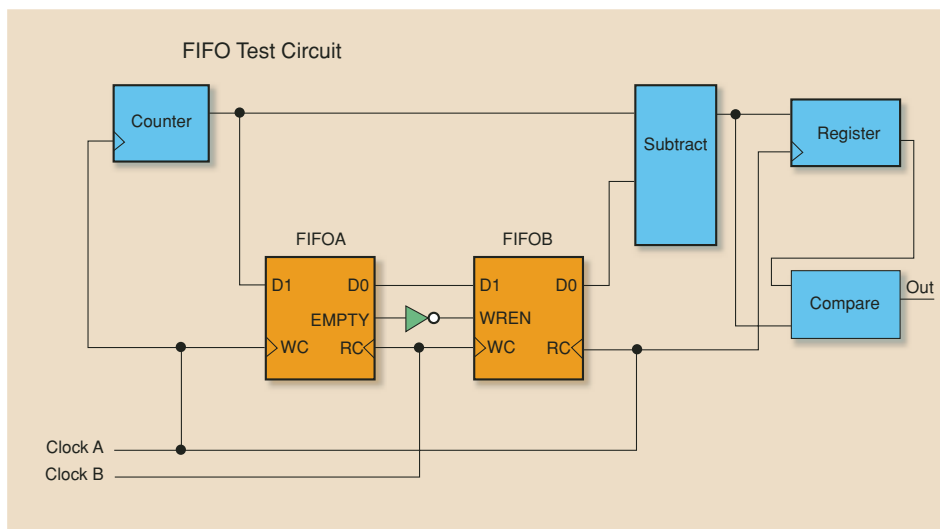


Figure 2 – FIFO test circuit

As the last data entry is being read, EMPTY goes high as a result of the read clock that reads the final data. You are supposed to disable the read operation until the EMPTY output has gone inactive again.

Note that both the rising and falling edge of the EMPTY status signal are made synchronous with the read clock, giving you a totally synchronous interface. If read clock enable stays active after the FIFO is empty, the read error flag is activated, but FIFO content and addressing are not disturbed.

ALMOST EMPTY and ALMOST FULL are programmable status outputs, available as a warning to slow down the read or write process, or as an indication of the data level in the FIFO (“dipstick”).

Implementation Details

Understanding FIFO design details is not necessary. It is all “under the hood,” and works without user intervention. But for the curious reader, let’s briefly explain.

Detecting FULL and EMPTY requires detecting identity of the write and read

Verifying the EMPTY Flag Synchronization

The only tricky detail in a FIFO with unrelated read and write clocks is the proper synchronization of the EMPTY and FULL flags that cross clock boundaries. Any design that might thus be exposed to metastability problems deserves special attention and scrutiny.

At Xilinx, we tested the EMPTY logic exhaustively by writing data into the FIFO at 200 MHz and reading it out at 500 MHz, which makes it go EMPTY soon after each write cycle (Figure 2). The detection logic was thus exercised, and the trailing edge of the EMPTY flag was re-synchronized to the write clock 200 million times a second.

More specifically, we wrote an ascending data sequence at 200 MHz and read it out at 500 MHz. We wrote the output data directly into a second FIFO at the same 500 MHz. We then read the second FIFO out at the original 200 MHz rate.

The combined dual FIFO forms a synchronous system, but with asynchronous data transfer between the two halves. When we synchronously subtracted the input data from the output data, the difference was constant, indicating flawless transfer at the 500 MHz read/write rate and no flag synchronization problem – even at this high rate.

When the two clock frequencies are uncorrelated, each read clock cycle has a different phase relationship with respect to the write clock. During any second, the active read clock edge steps across the ~5 ns write clock period in ~200 million different phase orientations, thus creating a timing granularity of 0.025 femtoseconds (one quadrillionth of a second). This resolution is millions of times better than any conventional deterministic test methodology can possibly achieve.

We ran this design for a whole week, with more than 10^{14} operations, without any error.



address pointers, which generally do not share a common clock. Binary counters would generate unacceptable glitches on the comparator output; using Gray-coded counters is the well-known solution to this problem.

The simplest way to build Gray counters is to start with a binary counter and synchronously convert its content into Gray code. The binary address counter values can then be used to calculate the programmable offset for detecting ALMOST FULL and ALMOST EMPTY.

Synchronization Issues

Because EMPTY can only be caused by a read operation, the leading edge is naturally synchronous with the read clock. But the trailing edge is caused by a write operation and is thus synchronous with the “wrong” clock. Moving the trailing edge of EMPTY over onto the read clock domain needs some flip-flops and invites the specter of metastability.

Virtex-4 FPGAs use a conservative synchronizer design that has been demonstrated to work reliably at a 500 MHz read clock rate. We ran a week-long test with ~200 and ~500 MHz asynchronous clock rates, generating EMPTY more than 10^{14} times without a single failure. The synchronizer delays the trailing edge of EMPTY by a few read clock periods. This latency is acceptable, since it does not affect top performance.

In a similar way, the trailing edge of FULL is synchronized to the write clock. The software default is for FULL to have one write clock latency. We therefore recommend using ALMOST FULL instead.

A well-designed FIFO buffer should never go FULL, and should go EMPTY only when you want to drain the last word from the buffer.

Conclusion

The hard-coded FIFO controller is available in every Virtex-4 block RAM, and uses no additional resources in the fabric. It also saves you from making any complex, time-consuming, and risky design decisions.

For a detailed description of the Virtex-4 FIFO controller, visit the Virtex-4 User Guide on the Xilinx website at www.xilinx.com/bvdocs/userguides/ug070.pdf.

Xcell journal

THE AUTHORITATIVE JOURNAL FOR PROGRAMMABLE LOGIC USERS

Would you like to write for Xcell Publications?

It's easier than you think.

We recently launched the Xcell Publishing Alliance to help you publish your technical ideas. We can help you – from concept research and development, through planning and implementation, all the way to publication and marketing.

Submit articles for our Web-based *Xcell Online* or our printed *Xcell Journal* and we will assign an editor and a graphics artist to work with you to make your work look as good as possible. Submit your book concepts and we will bring our partnership with Elsevier, the largest English language publisher in the world, and our broad industry resources to assist you in planning, research, writing, editing, and marketing.

For more information on this exciting and highly rewarding program, please contact:

Forrest Couch

Managing Editor, Xcell Publications

xcell@xilinx.com





Digital Clock Management in Virtex-4 Devices

The new Virtex-4 FPGA includes improvements and additions to the digital clock module.

by Ralf Kreuger
Sr. Staff Applications Engineer
Xilinx, Inc.
ralf.kreuger@xilinx.com

As FPGAs grow in size, quality on-chip clock distribution becomes increasingly important. Clock skew and clock delay impact device performance; managing clock skews and delays with conventional clock trees becomes more difficult in larger devices.

Xilinx® Virtex-4™ devices solve this challenge by providing as many as 20 fully dedicated on-chip digital clock management (DCM) circuits. DCM provides zero propagation delay and – along with fully differential global clock trees – low clock skew between output clock signals distributed throughout the device.

Each DCM can drive up to 12 of the 32 global clock routing networks within the device. The global clock distribution network minimizes clock skews due to loading differences. By monitoring a sample of the DCM output clock, the delay locked loop (DLL) compensates for the delay on the routing network, effectively eliminating the delay from the external input port to the individual clock loads within the device.

By taking advantage of DCM to remove on-chip clock delay, you can greatly simplify and improve system-level designs involving high fan-out, high-performance clocks.

In addition to providing zero delay with respect to a user source clock, DCM provides multiple phases of the source clock. The DLL can act as a clock doubler or divide the user source clock by up to 16.

DCM can also act as a clock mirror. By driving DCM output off-chip and back in again, you can use it to de-skew a board-level clock between multiple devices.

Digital Phase Shift (DPS)

Virtex-4 FPGAs provide a digital phase shift (DPS) module that phase shifts the DCM's output clock in small increments – 1/256th of its period. You can operate the versatile DPS in four different modes for maximum flexibility: fixed, variable-positive, variable-center, and direct.

Digital Frequency Synthesis (DFS)

The DCM digital frequency synthesis (DFS) module provides two outputs, CLKFX and CLKFX180, derived from the input clock by frequency multiplication and division. Through a frequency calculator, you provide the multiply and divide values implemented by the DFS module. For example, an M value of 19 and a D value of 8 yields a 2.375 source clock multiplier.

DCM Features

DCMs are located in the center column of the Virtex-4 architecture. This enables well-matched clock routes to and from every DCM for enhanced symmetry.

The Virtex-4 DCM's superior performance does not just include a wider operating range. It encompasses lower jitter, improved phase accuracy, finer phase-shift resolution, tolerance of imperfect clocks and board designs, less duty-cycle distortion, and less sensitivity to sporadic voltage changes.

Xilinx also added new features. You now

have the choice to trade off a wider phase shift range versus higher frequencies.

In addition, a new function in the Virtex-4 architecture is the dynamic reconfiguration port (DRP). The DRP allows you to directly access some features in DCM through a block RAM-style interface. You can directly phase shift the delay line elements and change M and D values.

The software view of DCM has changed as well. Three Virtex-4 primitives – DCM_BASE, DCM_PS, and DCM_ADV – offer progressive features to enhance your design choices.

Xilinx also added a new DCM companion block, the phase-matched clock divider (PMCD), to the Virtex-4 family. Let's discuss the clock management features of these new clock resources.

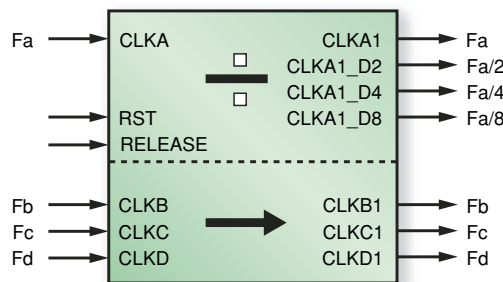


Figure 1 – Phase-matched clock divider

Phase-Matched Divided Clocks

PMCDs create as many as four frequency-divided and phase-matched versions of an input clock, CLKA. The output clocks are a function of the input clock frequency: divided-by-1 (CLKA1), divided-by-2 (CLKA1D2), divided-by-4 (CLKA1D4), and divided-by-8 (CLKA1D8).

CLKA1, CLKA1D2, CLKA1D4, and CLKA1D8 output clocks are rising-edge aligned to each other, but not to the input (CLKA). Figure 1 illustrates the new PMCD primitive.

Phase-Matched Delay Clocks

PMCDs preserve edge alignments, phase relations, or skews between the CLKA input clock and other PMCD input clocks. Three additional inputs (CLKB, CLKC, and CLKD) and three corresponding delayed outputs (CLKB1, CLKC1, and CLKD1) are available. The same delay is inserted to CLKA, CLKB, CLKC, and CLKD; thus, the delayed CLKA1, CLKB1, CLKC1, and CLKD1 outputs maintain edge alignments, phase relationships, or the skews of their respective inputs.

You can use PMCDs alone or with other clock resources, including global buffers and DCMs. Together, these clock resources provide flexibility in managing complex clock networks.

The PMCDs are located in the center column right next to the DCMs. They are grouped as pairs in each tile.

Conclusion

The many features and functions of the clock management subsystem allow you to maximize system performance. By taking advantage of DCM to remove on-chip clock delay, you can greatly simplify and improve system-level designs involving high fan-out, high-performance clocks. Virtex-4 devices have an abundance of clock management resources along with comprehensive software support.

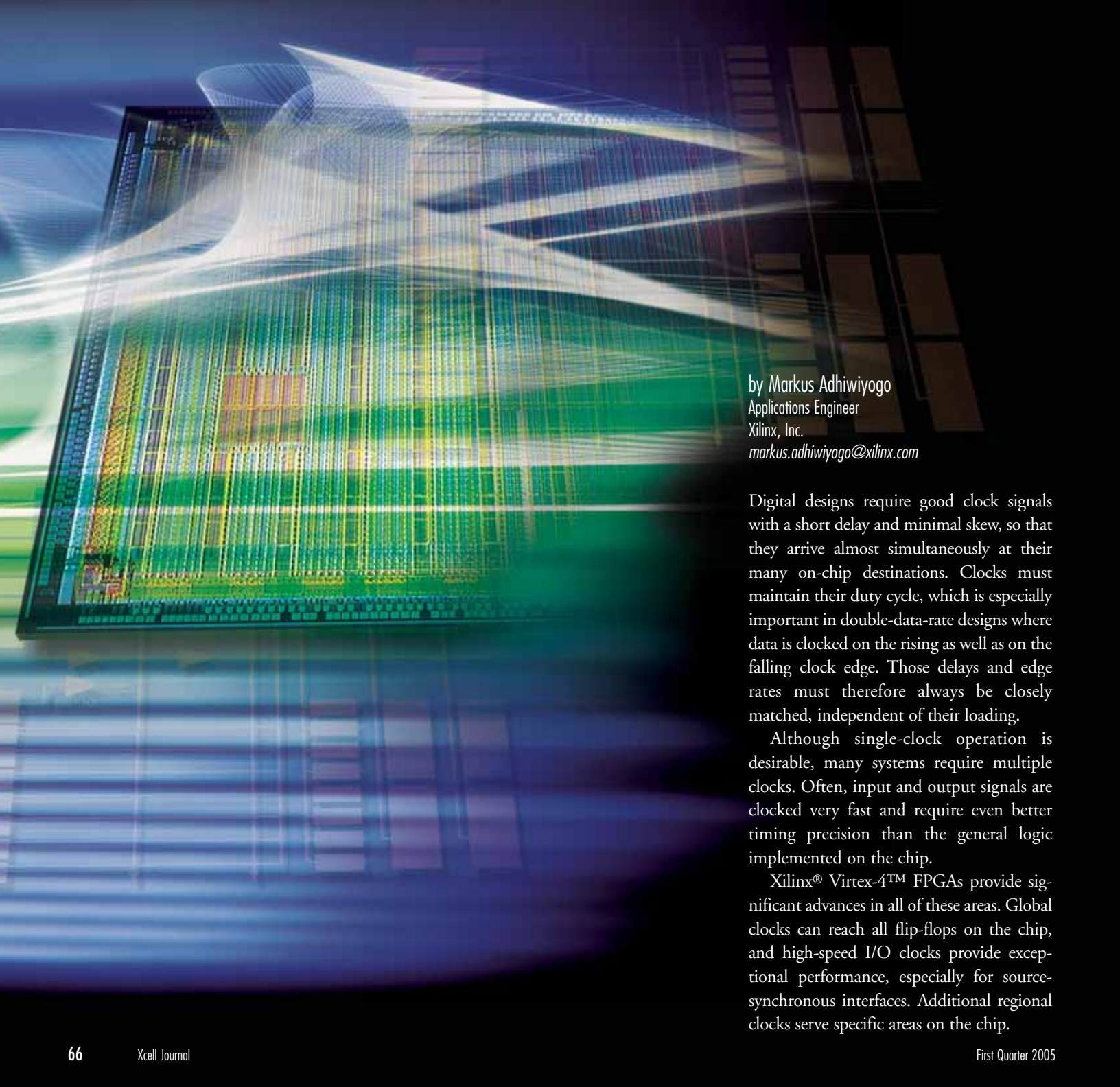
Specialized individual features further improve the ability to optimize design performance. Frequency synthesis is a powerful feature to generate a wide range of frequencies in the FPGA or the entire system. A fine-resolution phase-shift capability allows you to improve margins. And the new PMCD further increases the number of clock derivatives that can be generated without the use of additional DCMs.

For more information, see the user guide at www.xilinx.com/bvdocs/userguides/ug070.pdf.



Virtex-4 Clocking Resources

Xesium clocking networks are an innovative feature in Virtex-4 devices.



by Markus Adhiwiyogo
Applications Engineer
Xilinx, Inc.
markus.adhiwiyogo@xilinx.com

Digital designs require good clock signals with a short delay and minimal skew, so that they arrive almost simultaneously at their many on-chip destinations. Clocks must maintain their duty cycle, which is especially important in double-data-rate designs where data is clocked on the rising as well as on the falling clock edge. Those delays and edge rates must therefore always be closely matched, independent of their loading.

Although single-clock operation is desirable, many systems require multiple clocks. Often, input and output signals are clocked very fast and require even better timing precision than the general logic implemented on the chip.

Xilinx® Virtex-4™ FPGAs provide significant advances in all of these areas. Global clocks can reach all flip-flops on the chip, and high-speed I/O clocks provide exceptional performance, especially for source-synchronous interfaces. Additional regional clocks serve specific areas on the chip.

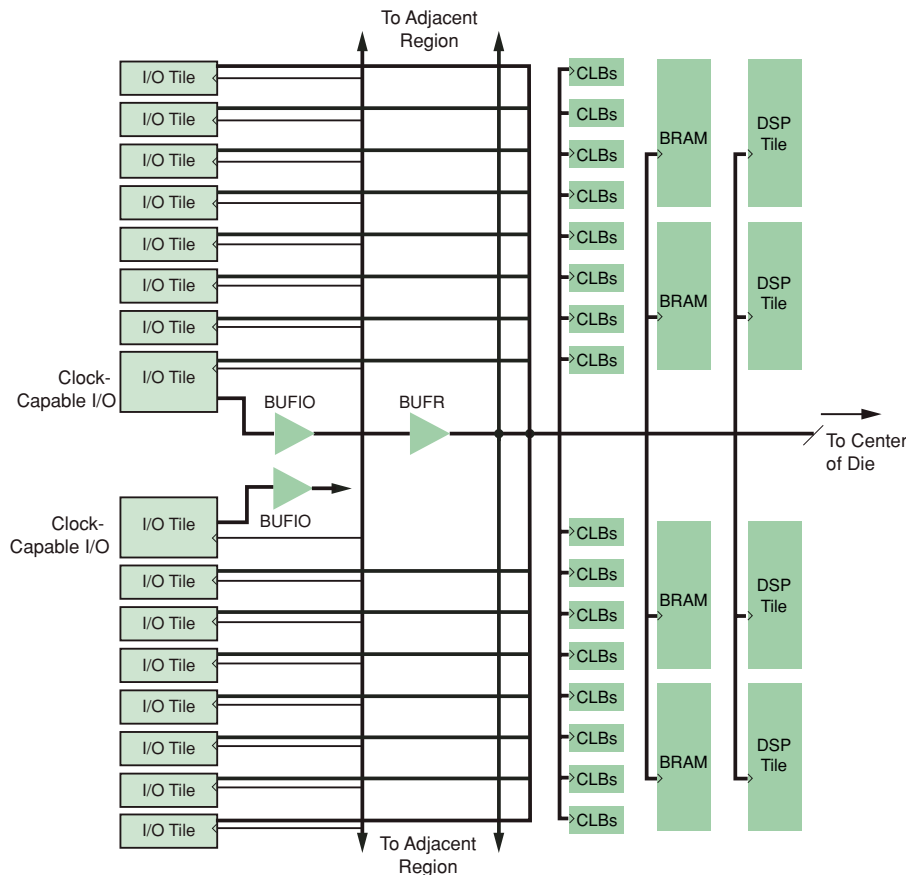


Figure 1 – BUFIO and BUFR clocking up to three regions

Clock Regions

For clocking purposes, each Virtex-4 device is divided into regions. The number of regions varies with device size, from 8 regions in the smallest device to 24 regions in the largest one.

Global Clocks

Independent of array size, each Virtex-4 FPGA has 32 low-skew global clock distribution networks that can each clock all sequential resources on the whole chip (CLBs, block RAMs, DCMs, and I/Os) and also drive logic signals. You can use any 8 of these 32 global clock lines in any region.

All global clock inputs have dedicated fast routing to the corresponding global clock buffer, which can also be used as a clock-enable circuit or a glitch-free multiplexer. It can select between two clock sources and can also switch away from a failed clock source – a new feature in the Virtex-4 architecture.

A global clock buffer is often driven by a

digital clock manager (DCM) to eliminate the clock distribution delay, or to adjust its delay relative to another clock. There are more global clocks than DCMs, and a DCM often drives more than one global clock.

Virtex-4 clock trees are designed for low skew and low power. Any unused branch is automatically disconnected. All global clock lines and buffers are implemented differentially. This minimizes duty-cycle distortion and improves common-mode noise rejection. The whole global clock network is designed for 500 MHz operation and beyond.

I/O Clocks and Regional Clocks

Virtex-4 devices have two additional clock types: I/O clocks and regional clock networks, two of each per region, used primarily for clocks forwarded into the Virtex-4 FPGA. I/O and regional clock networks are independent from the global clock networks, thus offering a maximum of 12 independent clock domains in any clock region.

Each clock region has two pairs of

clock-capable inputs, optimized for incoming high-frequency clocks. Clock-capable I/O pairs, like global clock inputs, are regular I/O pairs where the LVDS output drivers have been removed to reduce the input capacitance.

Each of these input pins or input pin pairs can connect to a BUFIO that drives a high-speed differential I/O clock network, which is dedicated to the I/O circuits and is ideally suited for source-synchronous data capture using the built-in serializer/deserializer (SerDes).

Each BUFIO can drive all I/O logic in its region as well as in the two adjacent regions (Figure 1). This means that one receive clock can control up to 47 differential or 95 single-ended receive data lines, ideal for many networking and memory interface applications.

Regional clocks form a third type of clock networks, each being able to span as many as three adjacent clock regions. Regional clocks drive single-ended nets and are intended for the parallel clock domain of the SerDes.

You can program the regional clock buffer to divide the incoming clock rate by any integer number from one to eight. This feature, in conjunction with the programmable SerDes in the I/O block, allows source-synchronous systems to cross clock domains without using additional logic resources.

Conclusion

Virtex-4 clocking resources have been optimized for high clock rates and multiple clock domains. Thirty-two global clock networks provide high-performance clocking across the whole chip, with short delay, low skew, and stable duty cycles.

Many localized clock networks serve the I/O for high-speed source-synchronous applications. These clock networks are used in conjunction with the built-in SerDes and reduce the burden on global clock resources.

Last but not least, all of these resources are easy to use. They are automatically handled by the Xilinx ISE 6.3i software.

For more information, visit www.xilinx.com/products/virtex4/capabilities/xesium.htm.



Alpha Blending Two Data Streams Using a DSP48 DDR Technique

Achieve full throughput of the DSP48 slice with a double-data-rate technique.

by Reed Tidwell
Sr. Staff Applications Engineer
Xilinx, Inc.
reed.tidwell@xilinx.com

The XtremeDSP™ system feature, embodied as the DSP48 slice primitive in the Xilinx® Virtex-4™ architecture, is a high-performance computing element operating at an industry-leading 500 MHz. The design of the Virtex-4 infrastructure supports this rate, with Xesium clock technology, Smart RAM, and LUTs configured as shift registers.

Many applications, however, do not have data rates of 500 MHz. So how can you harness the full computing performance of the DSP48 slice with data streams of lower rates?

The answer is to use a double-data-rate (DDR) technique through the DSP48 slice. The DSP48 slice, operating at 500 MHz, can multiplex between two data streams, each operating at 250 MHz.

One application of this technique is alpha blending of video data. Alpha blending refers to the combination of two streams of video data according to a weighting factor, called alpha. In this article, we'll explain the techniques and design considerations for applying DDR to two data streams through a single DSP48 slice.

All Virtex-4 devices have DSP48 slices, although the SX family contains the largest number (an industry-high 512) and the highest concentration of DSP48 slices to logic elements, making it ideal for math-intensive applications ...

Virtex-4 DSP48

The DSP system elements of Virtex-4 FPGAs are dedicated, diffused silicon with dedicated, high-speed routing. Each is configurable as an 18 x 18-bit multiplier; a multiplier followed by a 48-bit accumulator (MACC); or a multiplier followed by an adder/subtractor. Built-in pipeline stages provide enhanced performance for 500 MHz throughput – 35% higher than for competing technologies.

All Virtex-4 devices have DSP48 slices, although the SX family contains the largest number (an industry-high 512) and the highest concentration of DSP48 slices to logic elements, making it ideal for math-intensive applications such as image processing.

A triple-oxide 90 nm process makes the DSP48 slice very power-efficient.

Architectural features, including built-in pipeline registers, accumulator, and cascade logic nearly eliminate the use of general-purpose routing and logic resources for DSP functions, and further reduce power. This slashes DSP power consumption to a fraction when compared to Virtex-II Pro™ devices.

DDR with Two Data Streams

DDR, in this context, refers to multiplexing two input data streams into one stream at twice the rate, interleaving (in time) the data from each stream (Figure 1). Figure 1 also shows the reverse operation, creating two parallel resultant streams after processing.

You can drive the DSP48 slice inputs at the fast 500 MHz clock rate from CLB

flip-flops; CLB LUTs configured as shift registers (SRL16); or directly from block RAM. Block RAM, configured as a FIFO using the built-in FIFO support, also supports the 500 MHz clock rate.

Design Considerations

Dealing with data at 500 MHz requires great care; you should observe strict pipelining with registers on the outputs of each math or logic stage. The DSP48 slice provides optional pipeline registers on the input ports, on the multiplier output, and on the output port from the adder/subtractor/accumulator. Block RAM also has an optional output register for efficient pipelining when interfaced to the DSP48 slice.

Where you are using CLBs, place only minimal levels of logic between registers to provide maximum speed. For DDR operation, only a 2:1 mux (a single LUT level) is required between pipeline stages. Whether you are interfacing to the DSP48 slice with memory or CLBs, placing connected 500 MHz elements in close proximity minimizes connection lengths in the general routing matrix.

DDR requires the DSP48 slice to operate at double the frequency of the input data streams. You can use a DCM to provide a phase-aligned double-frequency clock using the CLK 2X output.

Another aspect of inserting DDR data through a section of pipeline is ensuring that data passes cleanly between clock domains. This may require adding extra registers clocked with the double-frequency clock at the output of the double-pumped section, to synchronize the data with the original clock. The rule of thumb is that in order to insert a double-pumped section cleanly into a single-pumped pipeline, there must be an even number of register delays in the double-pumped section.

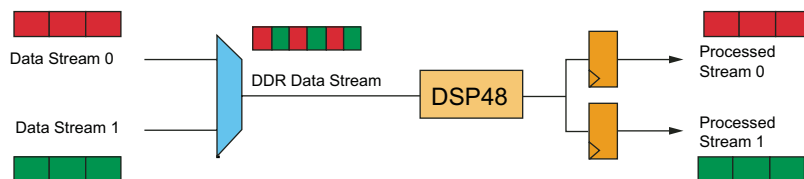


Figure 1 – DSP48 DDR

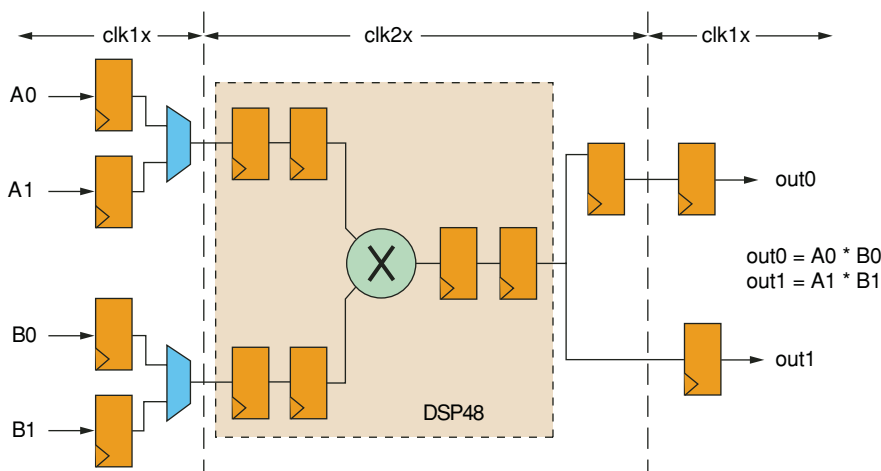


Figure 2 – Two-stream multiply through DSP48 slice

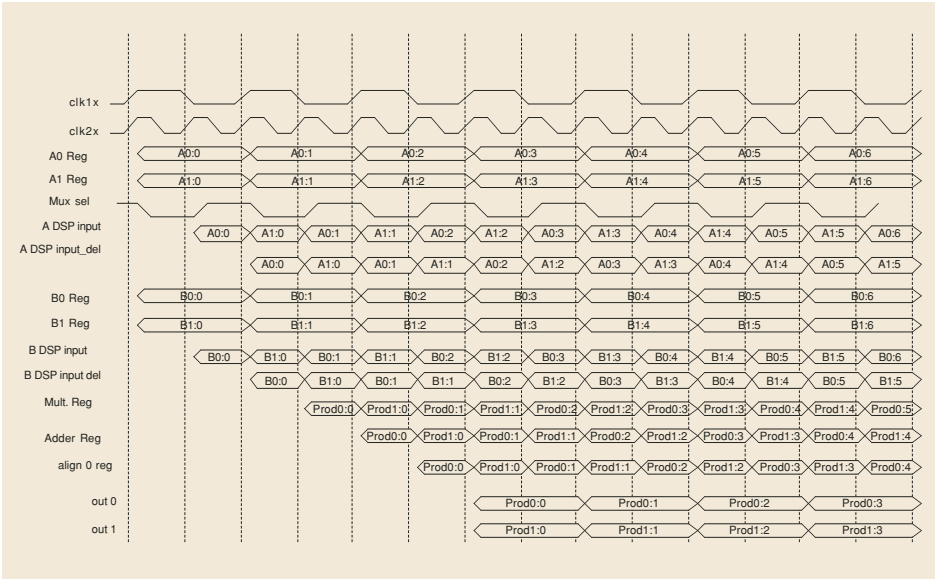


Figure 3 – Timing of two-stream multiply

Implementation

Several configuration options exist for implementing DDR functionality. Figure 2 shows a straightforward implementation.

In Figure 2, stream 0 consists of A0 and B0 inputs. We multiply them together and output as out0. Likewise, stream 1 consists of inputs A1 and B1 multiplied together and output as out1. There are two clock domains: the clk1x domain, at the nominal data stream frequency, and the clk2x domain, at twice the nominal frequency.

Figure 2 shows two registers after the multiplier. The second is the accumulation register, even though we do not use accumulation in this configuration. The register, however, is still required to achieve the full, pipelined performance. We use two sets of registers on the inputs of the DSP to make the total delay through the DSP48 slice an even number (four) for easier alignment of the output data with clk1x. These registers are “free” because they are built into the DSP48 slice, and using them reduces the need for alignment registers external to the DSP48 slice. The extra pipeline register on out0 compensates for taking stream 0 into the DSP one clk2x cycle before stream 1. As seen from the timing diagram in Figure 3, this is required to realign the stream 0 data back into the clk1x domain.

Note that the input mux select, mux_sel, is essentially the inverse of clk1x. It is important, however, to generate this signal from a register based on clk2x (rather than deriving it from clk1x) to avoid hold-time violations on the receiving registers.

At the transitions between clock domains, the data have only one clk2x period to set up. This is the reason to have no

logical operations between registers in the two domains. The placement of the first registers in the clk1x domain is more critical than other registers in the same domain.

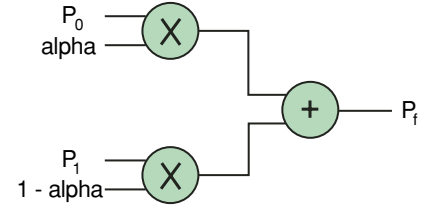


Figure 4 – Alpha blend formula in graphical terms

Alpha Blending

Alpha blending of video streams is a method of blending two images into a single combined image, such as fading between two images, overlaying anti-aliased or semi-transparent graphics over an image, or making a transition band between two images on a split-screen or wipe. Alpha is a weighting factor defining the percentage of each image in the combined output picture. For two input pixels

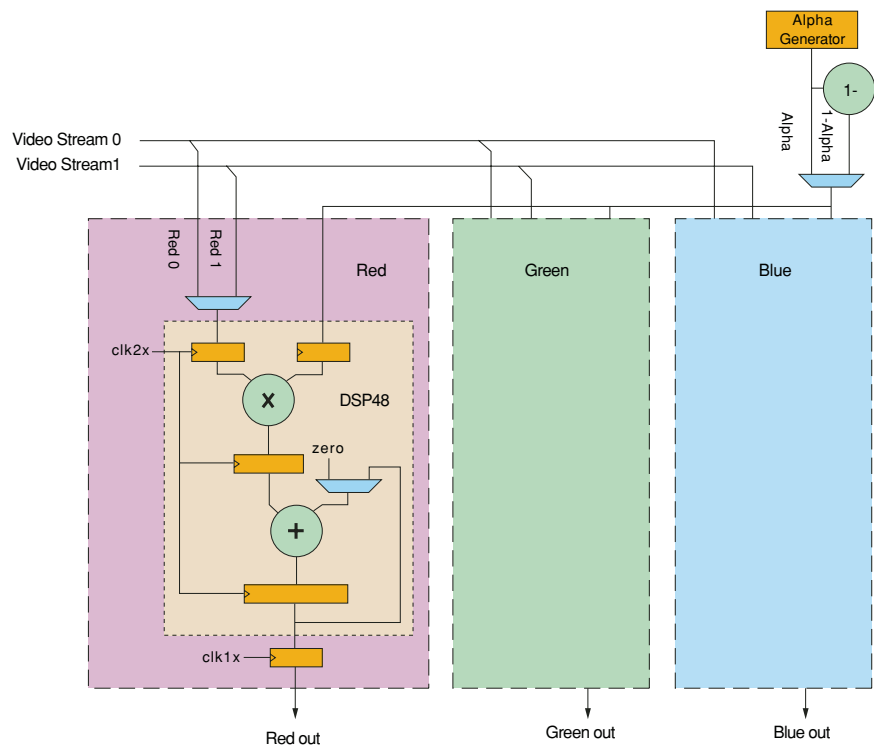


Figure 5 – Alpha blend on three-component video

You can efficiently use the high-performance of Virtex-4 devices with DSP48 slices by processing multiple data streams in a time-multiplexed fashion.

(P0, P1, and a blend factor, α , where $0 \leq \alpha \leq 1.0$), the output pixel P_f will be:

$$P_f = \alpha P_0 + (1-\alpha)P_1 \text{ (see Figure 4)}$$

This operation is performed separately for each component: red, green, and blue.

A pixel rate of 250 MHz or less is sufficient for all standard and high-definition video rates, and common Video Electronics Standards Association (VESA)

standards as high as 1600 x 1200 at 85 Hz. Therefore, one DSP48 slice can perform the multiply and add on one component, and a set of three slices can alpha blend the three components from each of two video streams, as shown in Figure 5. The operations must be performed identically and in parallel on each of the three components.

There are several ways to implement alpha blending depending on the nature

of the video streams and how alpha is generated. Figure 6 shows a basic implementation with two video streams alternating as one multiplier input. The other multiplier input alternates between alpha and 1-alpha.

The operating mode of the adder alternates between add zero (pass through) mode and add output (accumulate) mode. The DSP48 slice output register contains the result of the Video0 * alpha multiply during one clock cycle, and the final result (Video1 * (1 - alpha) + Video0 * alpha) on the alternate clock. Figure 7 shows the timing for this configuration.

The align registers on the inputs of the DSP are used to make the total delay through the DSP48 slice an even number (four), as explained in the previous example. The final output register for blend loads new data to every other DSP clock to register the blend results at the original pixel rate.

Conclusion

You can efficiently use the high-performance of Virtex-4 devices with DSP48 slices by processing multiple data streams in a time-multiplexed fashion. With careful design, a single DSP48 can perform multiply operations on two independent data streams, operating at 250 MHz each.

Alpha blending of video streams, as outlined in this article, is one example of processing two data streams through a single DSP48 slice. This capability complements the DSP features of Virtex-4 FPGAs – including built-in pipelining and cascading, integrated 48-bit accumulator, and an abundance of DSP48 slices in the SX family – to make Virtex-4 devices the ideal DSP platform.

For details about the DSP48 slice, refer to the “Virtex-4 FPGA Handbook,” Chapter 10, or the “XtremeDSP Design Considerations User Guide” at www.xilinx.com/bvdocs/userguides/ug073.pdf.

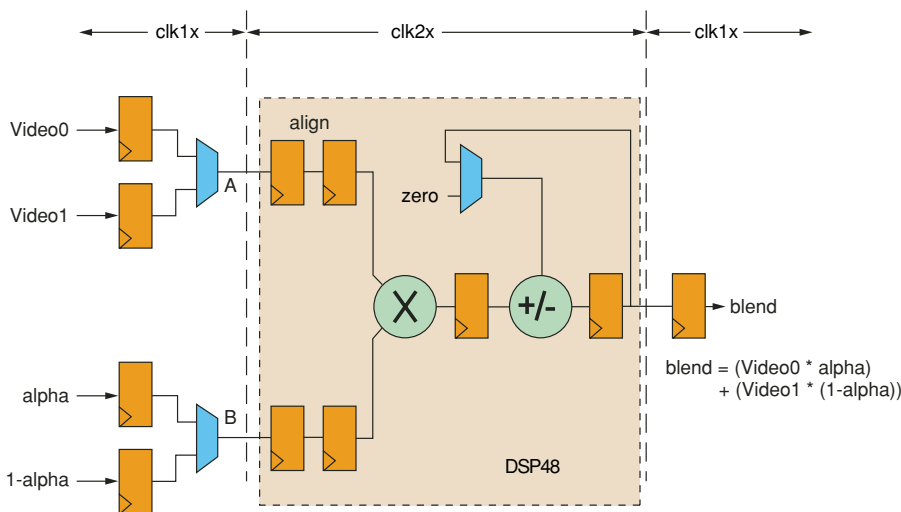


Figure 6 – Alpha blend implementation (one component)

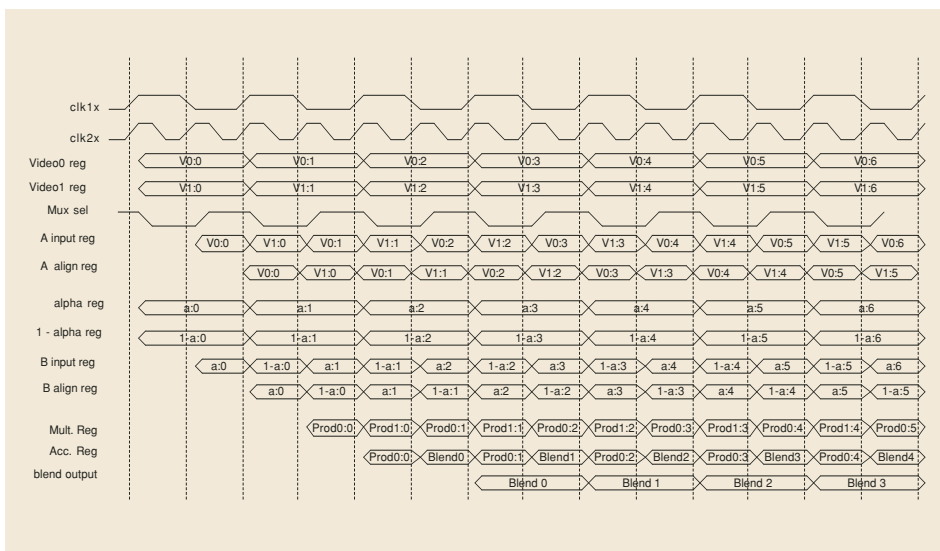


Figure 7 – Alpha blend timing



Dynamic Phase Alignment with ChipSync Technology in Virtex-4 FPGAs

ChipSync technology built into every I/O supports dynamic phase alignment solutions for high-speed source-synchronous interfaces.

by Tze Yeoh
Product Applications Engineering
Xilinx, Inc.
tzeyi.yeoh@xilinx.com

Xilinx® FPGAs provide connectivity in very high speed source-synchronous bus interfaces. Transmission rates of 1 Gbps and higher are not uncommon for these types of interfaces.

In source-synchronous interfaces, the transmitter forwards a dedicated clock along with the data. As data rates skyrocket to 1 Gbps and beyond, you may find that your timing budgets are eaten away by skew and jitter.

Skew is defined as the difference in arrival time between signals sent at the same time. It is caused by variations in board trace lengths, connectors, package flight-time delays, and secondary parasitic effects. Figure 1 illustrates how the improper routing of board traces and the use of connectors contributes towards skew at the receiver.



The ISERDES built into every I/O contains a dedicated serial-to-parallel converter that converts the high-speed serial stream to a sequence of parallel words that can be processed at a much slower rate within the FPGA.

Another challenge is jitter, the deviation from ideal timing caused mostly by slow transition times, ground bounce, inter-symbol interference, and electromagnetic interference. Figure 2 illustrates the combined effects of skew and jitter on a system designer's timing budget.

In a real system, many bits of data (16, for example) are received in parallel and must be clocked into the receiver by the common clock sent together with the data. Ideally, the clock edge arrives in the middle of the bit time, thus offering a maximum timing margin.

But in reality, the individual data bits arrive at slightly different times, and each suffers from timing jitter on its rising and falling edges, and therefore the clock signal also suffers from timing jitter. All of these effects combine to limit the data-valid window, and thus might lead to unreliable data transmission.

Virtex-4™ data and clock inputs offer ChipSync™ technology, facilitating dynamic phase alignment (DPA). DPA can greatly reduce the skew between different data lines, as well as between the data lines and their associated clock input.

Using a system-generated training pattern, the receiving FPGA can adjust the input delay of each data and clock input, using individual precision delay lines on every input buffer. Gross errors exceeding one bit time pass through the bit-serial interface, but can be corrected after serial-to-parallel conversion using the Bitslip module.

A Generic Networking Interface Example

The generic interface is defined by a 16-channel bus and a forwarded clock. The signaling standard is low-voltage differential signaling (LVDS). The interface protocol specifies a de-skewing method called “train-

the delay lines in a region are continuously being calibrated by a servo system using a dedicated delay line, a 200 MHz user-provided clock, and a phase-comparator-driven PLL circuit that adjusts the delay line(s) such that the 64-stage delay equals one period of the clock ($5 \text{ ns} / 64 = 78 \text{ ps}$ per tap).

All delay lines in one region share a common adjustment, and thus have the same tap delay, as accurately as delay tracking in a small silicon area allows. The reference frequency is specified, tested, and supported by software at 200 MHz. Minor variations can be tolerated, and jitter is filtered out by the control structure. This programmable precision delay will find its way into many innovative applications. Here it is described only as a method to achieve dynamic phase alignment.

The ChipSync technology built into every I/O contains a dedicated serial-to-parallel converter that converts the high-speed serial stream to a sequence of parallel words that can be processed at a much slower rate within the FPGA. This feature decouples the high-speed serial data transfer from the clock rate supported by the FPGA fabric.

The converter supports both single data rate (SDR) and double data rate (DDR) modes. In SDR mode, the serial-to-parallel converter is fully programmable to generate anywhere from 2- to 8-bit parallel words. In DDR mode, the converter can be programmed to de-serialize by a factor of 4, 6, 8, or 10, as specified by the HDL attributes of the ChipSync technology. The maximum width in a single ChipSync module is six. For larger bit widths, you can connect two adjacent ChipSync modules in master-slave mode.

Word alignment can correct for data skew greater than one bit period by comparing the parallel version of the incoming pattern to the pre-specified training pattern. The Bitslip module enables you to

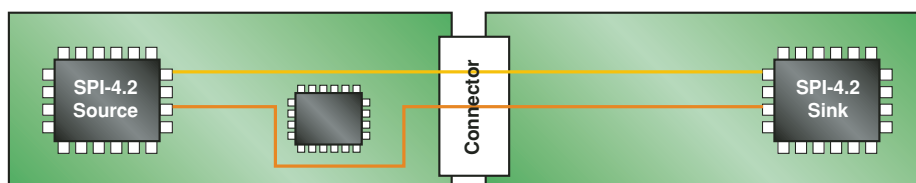


Figure 1 – Improper board trace routing and use of connectors contribute to skew

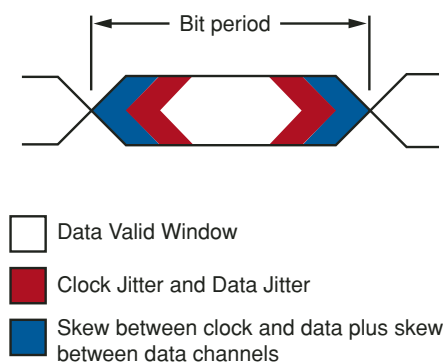


Figure 2 – Effects of skew and jitter on timing budgets

ing.” During the initialization phase, the transmitter sends a repetitive 20-bit training pattern. The receiver uses it to de-skew the interface by delaying each data bit such that it is optimally centered over the received clock edge. The interface specification calls for the receiver to correct data skew as much as ± 1 bit time of channel-to-channel skew.

This fine-grained delay adjustment uses a 64-tap delay line with a counter-controlled tap multiplexer available on each input. All of

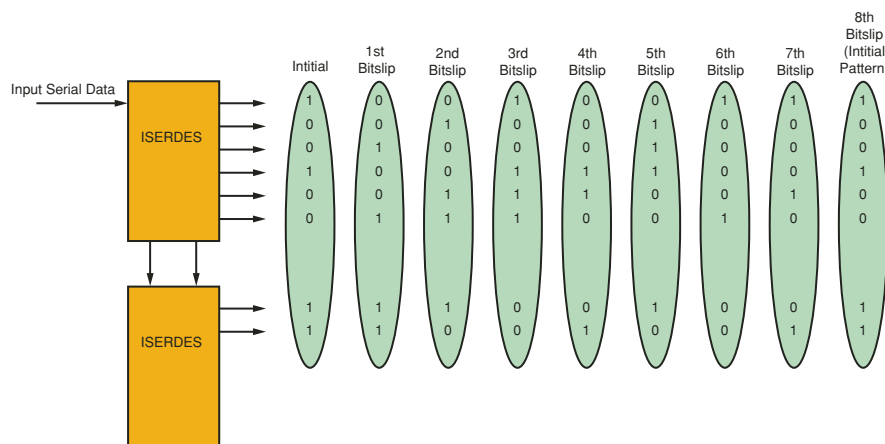


Figure 3 - Operation of Bitslip

match an incoming data stream to a pre-determined data pattern by shifting the output of the dedicated serial-to-parallel converter. An example of this feature in operation is given in Figure 3.

The IDELAY, SERDES, and Bitslip features are encapsulated in a module called ISERDES, available as part of the ChipSync technology in every single I/O.

The Virtex-4 DPA Solution

Let's use the Virtex-4 ChipSync technology features previously described to create a DPA solution that meets interface requirements. There are three basic steps in the solution:

- Bit alignment – completed during the initialization procedure, its purpose is to correct for skew less than one bit time and position the clock edge at the center of the data eye
- Word alignment – completed during the initialization procedure, its purpose is to align the incoming data stream to the pre-determined training pattern
- Real-time window monitoring – continuously monitors the data eye so that the clock edge is always centered to the data eye

Figure 4 illustrates the implementation of DPA in a Virtex-4 device.

The goal of the bit-alignment procedure is to position the captured clock edge in the center of the data eye to provide maximum margin. The bit-alignment procedure takes advantage of the dedicated 64-tap delay line feature of the ISERDES.

The word alignment procedure aligns the output pattern from the ISERDES to a specific training pattern. This procedure effectively removes word skew and aligns all channels to a specific word boundary. The word alignment unit primarily uses the Bitslip module of the ISERDES. Each channel monitors the pattern streaming in. If the training pattern is not found, activate Bitslip until the pattern is found. Once found, the channel is – by definition – de-skewed.

After the initialization stage using the training procedure, the channels are assumed to remain trained throughout

normal operation. However, the data valid window might shift because of operating conditions. The window monitoring unit can continuously monitor the data valid window during normal operation and can adjust the sampling point as necessary to provide maximum margin.

Conclusion

Dynamic phase alignment is a critical function in many bus interfaces as data rates explode into the gigabit range. As FPGAs are increasingly being used directly in the data path of these very high speed interfaces, dynamic phase alignment in the FPGA is a must.

Virtex-4 ChipSync technology built into every I/O enables you to quickly and easily develop a DPA solution that meets your application.

An application note describing the implementation of DPA is available at www.xilinx.com/bvdocs/appnotes/xapp700.pdf. The application note, "Dynamic Phase Alignment for Networking Applications," is published as XAPP 700. The reference design enables you to quickly understand how to implement a DPA solution that fits your particular application. ●●●

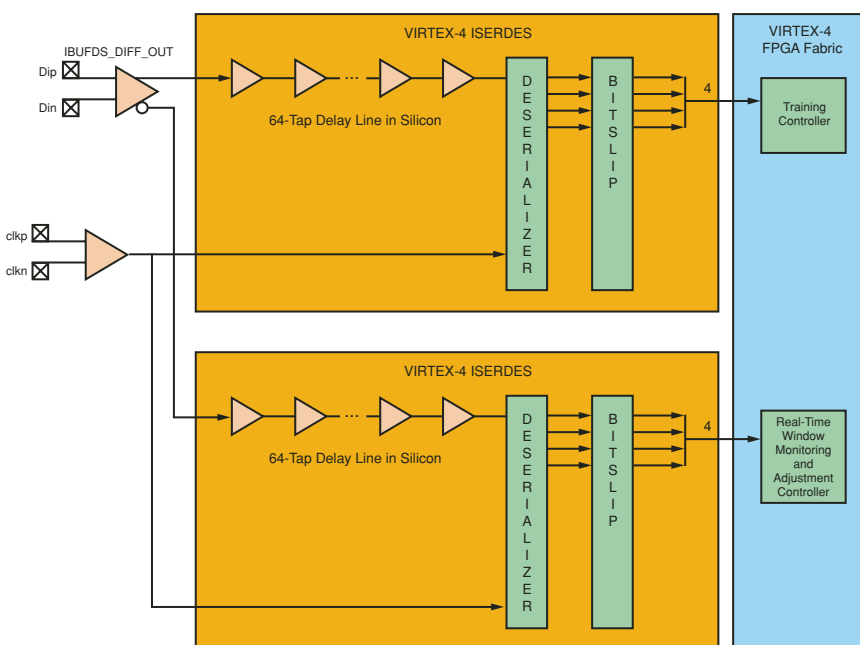
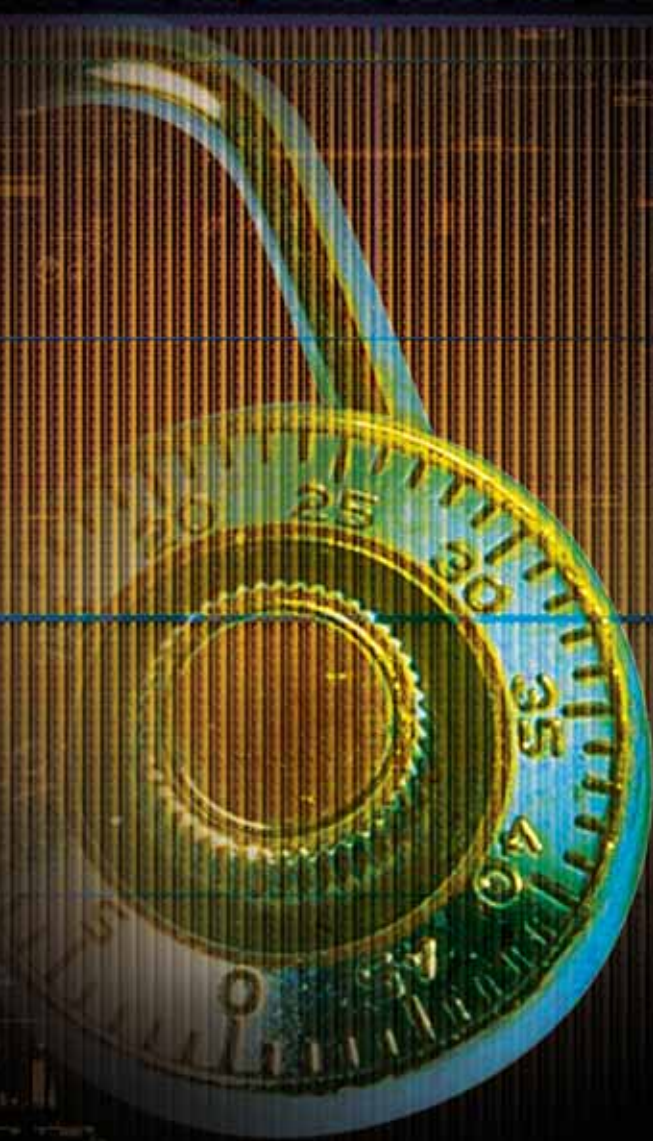


Figure 4 - Virtex-4 DPA implementation with ChipSync technology



Lock Your Designs with the Virtex-4 Security Solution

Virtex-4 FPGAs provide an up-to-date AES encryption scheme to prevent IP or microchip design theft.



by Chen Wei Tseng
Configuration Product PAE
Xilinx, Inc.
chenwei.tseng@xilinx.com

Because of the necessary configuration of FPGAs on each power up, as their popularity increases so do design security concerns. Without proper protections, attackers could easily clone or reverse-engineer the bitstream during FPGA configuration.

All Xilinx® Virtex-4™ devices have an on-chip decryptor that can be enabled to make the configuration bitstream secure. Virtex-4 has implemented the Advanced Encryption Standard (AES) scheme for securing the bitstream.

Modern Security Design

Xilinx has replaced the Triple DES encryption scheme implemented in the Virtex-II™ architecture with AES. Although both encryption schemes provide a high level of security, AES offers both increased security and throughput over Triple DES by replacing three 56-bit keys with one 256-bit key and allowing configuration clocking frequencies as high as 100 MHz.

Let's review some key benefits of the Xilinx Secure Chip solution.

1. AES is an official government standard, FIPS-197, supported by the National Institute of Standards and Technology and the U.S. Department of Commerce. The NSA has also certified AES' ability to protect classified communication to the top secret level.
2. The AES key can only be programmed through the JTAG interface. This allows you to monitor any unwanted activities on the JTAG lines both externally and internally with the BSCAN_Virtex4 primitive.
3. A battery-backed volatile key provides the maximum protection against hostile hacking.
4. This low-cost solution includes widely available standard components such as a Rayovac™ lithium battery.
5. Encryption key storage (Figure 1) has a long life span (20+ years).

Advanced Encryption Standard (AES)

Although the Triple DES algorithm remains effective against attacks, AES is now replacing DES in many applications as the most secure encryption scheme. As specified by FIPS-197, AES has the NSA-approved cryptographic algorithm that can be used to protect electronic data.

AES employs a cipher block that eliminates symmetry in the behavior of the cipher to overcome shortcomings of the DES' key. The non-linearity of the AES key expansion practically eliminates the possibility of equivalent keys.

Because of its key strength, AES is suited for applications such as banking, defense, government, and sophisticated technical applications such as ATM, HDTV, broadband ISDN, voice, and satellite.

Data Encryption Support

The Virtex-4 AES system comprises software-based bitstream encryption and on-chip AES (Rijndael) decryption with cipher block chaining (CBC) to decrypt the incoming bitstream. The AES key is stored in dedicated memory, powered by

either an auxiliary power supply (V_{CCAUX}) or an externally connected battery.

To combat a brute-force software attack such as key search, Virtex-4 devices feature a 256-bit AES key system that enables 1.1×10^{77} possible key combinations. To program the key, the device must enter "key-access mode" in IEEE1532 flow via JTAG. Once in this mode, the previous encryp-

configuration interface as SelectMAP to access configuration logic internally so that you can partially reconfigure the device for extra design security.

In addition to ICAP, Virtex-4 devices can monitor activities on the external JTAG pins with the internal BSCAN_Virtex4 primitive. The BSCAN_Virtex4 primitive mirrors the activity on the TDI pin and

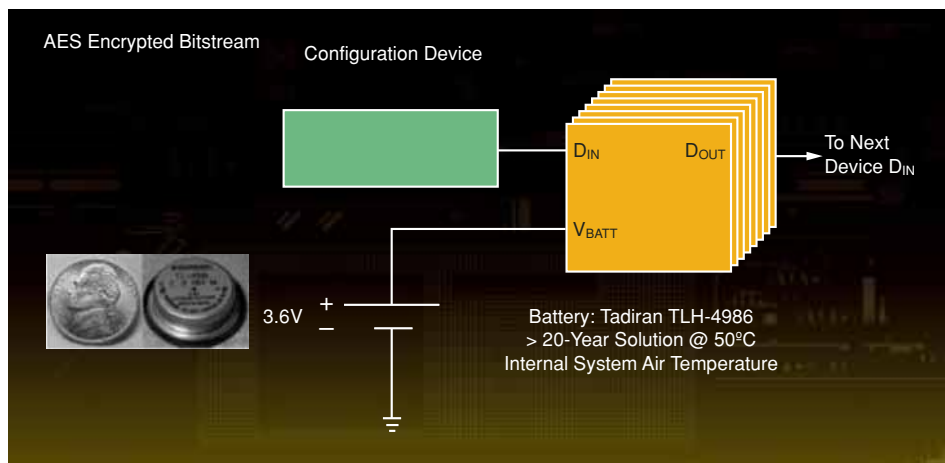


Figure 1 – Encrypted bitstream reference circuit for system-level applications

tion key will be cleared to prevent readback of the key. (Further flow details are documented in the Virtex-4 1532 BSDL files.) If the encryption keys are compromised, you can update the design with new keys and new encrypted bitstreams.

Virtex-4 FPGAs also embed the memory holding the key under layers of metal. Because the key is stored in volatile memory, disrupting the power supply for the key memory during hardware attacks will result in key loss.

You can always use a non-encrypted bitstream to configure the device regardless of the presence of the key. For example, when loading a non-encrypted bitstream, you should be careful when generating the bitstream. The proper security level should be set if you want readback of the non-encrypted bitstream. Reconfiguring the encrypted bitstream, however, would require you to toggle the PROG pin, cycle power, or issue one of two JTAG instructions: JPROG or JSTART.

Internally, you can use the internal configuration access port (ICAP) to reconfigure the device. ICAP provides the same

outputs several JTAG tap controller states, such as Test-Logic-Reset or Update-DR. Tampering with the JTAG during a "side channel" attack can be detected. You can then take countermeasures such as cutting power to the FPGA – including V_{BATT} – or erasing and writing a new encryption key by once again entering the key access mode.

Moreover, you can return any faulty part to Xilinx for testing without having to provide the encryption key for returned material analysis.

Software Integration

Xilinx ISE™ version 7.1i will have full software support for encrypted bitstream and key creation. Generating an encrypted bitstream requires only two additional bitgen options. For example, "bitgen -g encrypt: yes -g key0:AA995566 top.ncd top.bit" will automatically create an encrypted bitstream (top.bit) and the encryption key (top.nky) with the key of "AA995566." You must then load the top.nky file into the device through the JTAG interface before loading the encrypted bitstream.



Designing secure systems incorporating batteries for volatile storage is a proven method in multiple markets that is recognized as the highest form of security...

As for the GUI, Xilinx Project Navigator offers encryption options in the Generate Programming File command. You can set preferences for allowing readback, partial reconfiguration, and encryption.

iMPACT, the Xilinx programming tool, allows you to program just the key or the encrypted bitstream with the key. For independent programming applications, the detailed steps to download the encryption key are documented in the Virtex-4 IEEE1532 BSDL files, which are installed in the Xilinx/Virtex4/data directory with ISE installation, or downloadable from www.xilinx.com/support/sw_bsdll.htm.

Battery-Secured Systems

Designing secure systems incorporating batteries for volatile storage is a proven method in multiple markets that is recognized as the highest form of security and is required by the U.S. government for its secured modules (<http://csrc.nist.gov/publications/fips/fips140-2/fips1402.pdf>).

Several misconceptions exist related to battery use – some believe that batteries will require additional maintenance cycles. These fears are unfounded: maintenance and lifetime are of no concern for most applications, and the lifetime of the battery will usually far exceed the useful lifetime of the product.

All batteries “self discharge” when sitting idle, even with no load. Modern lithium batteries feature extremely low self-discharge rates. Rayovac lithium batteries self-discharge at a rate of less than 0.3% per year. Even at higher temperatures, the self-discharge experiences only very minor deterioration – in this example, let’s use a conservative 0.6%. The capacity of the BR1225 is 50 mAh.

Assume that the Virtex-4 IBATT current value is 50 nA. The VBATT signal is routed internally to the PCB to eliminate leakage currents. The self-discharge per hour is 34 nA.

$$34 \text{ nA} + 50 \text{ nA} = 84 \text{ nA}$$

$$50 \text{ mAh} / 0.000084 \text{ mA} = 595238 \text{ hours} = \sim 67 \text{ years}$$

Thus, a 20-year product life is easily achieved using a battery.

For more information about battery life expectancy calculations and design considerations, see Xilinx XAPP766, “Using High Security Features in Virtex-II Series FPGAs,” at <http://www.xilinx.com/bvdocs/appnotes/xapp766.pdf>.

Conclusion

Virtex-4 devices provide the most up-to-date security option for your designs. With the ease of integrated software flow, minimal board space requirements, and maximum security through AES, the Virtex-4 Secure Chip AES security solution is ideal for keeping hackers from your designs.

For more information about the Advanced Encryption Standard, please visit:

- <http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf>
- <http://csrc.nist.gov/encryption/aes/rijndael/>
- <http://csrc.nist.gov/encryption/aes/rijndael/Rijndael.pdf>
- <http://csrc.nist.gov/encryption/aes/>
- <http://csrc.nist.gov/encryption/aes/round2/r2report.pdf>
- <http://csrc.nist.gov/encryption/aes/round2/NSA-AESfinalreport.pdf>

Let Xilinx help you get your message out to thousands of programmable logic users worldwide.

That’s right ... by advertising your product or service in the *Xilinx Xcell Journal*, you’ll reach more than 70,000 engineers, designers, and engineering managers worldwide.

The *Xilinx Xcell Journal* is an award-winning publication, dedicated specifically to helping programmable logic users – and it works.

We offer affordable advertising rates and a variety of advertisement sizes to meet any budget!

Call today :
(800) 493-5551

or e-mail us at
xcelladsales@aol.com

*Join the other leaders
in our industry and advertise
in the Xcell Journal!*



Dynamic Reconfiguration of Functional Blocks

The Virtex-4 dynamic reconfiguration port offers an innovative way to reprogram functions in the FPGA.

by Ralf Krueger
Sr. Staff Applications Engineer
Xilinx, Inc.
ralf.krueger@xilinx.com

Configuration memory in Xilinx® Virtex™ FPGAs is used primarily to implement user logic, connectivity, and I/Os, but it is also used for other purposes. For example, it specifies a variety of static conditions in the two functional blocks, DCMs and RocketIO™ multi-gigabit transceivers (MGTs).

Sometimes an application requires a change in the conditions of the functional blocks while the blocks are operational. You can accomplish this through the global internal configuration access port (ICAP) or through partial dynamic reconfiguration, using JTAG or SelectMap in the “persist” mode.

Since the late 1990s, all Virtex FPGAs have supported this powerful dynamic partial reconfiguration feature. However, partial dynamic reconfiguration required you to have a detailed knowledge of the configuration logic functions, the configuration registers, and the location of configuration bits.



DRP Functionality

The new Virtex-4™ dynamic reconfiguration port (DRP) is an integral part of the two functional blocks, as it simplifies the dynamic reconfiguration process greatly. These configuration ports exist in the DCMs and RocketIO MGTs.

In this article, we'll describe the addressable, parallel write/read configuration memory implemented in each functional block that permits reconfiguration. This memory has the following attributes:

- It is directly accessible from the FPGA fabric. Configuration bits can be written to and/or read from depending on their function.
- Each bit of memory is initialized with the value of the corresponding configuration memory bit in the bitstream. Memory bits can also be changed later using the ICAP.
- The output of each memory bit drives the functional block logic, so the content of this memory determines the configuration of the functional block.

The address space can include status (read-only) and function enables (write-only). Read-only and write-only operations can also share the same address space. Figure 1 shows how the bitstream configuration bits drive the logic in functional blocks and how the reconfiguration logic changes the flow to read or write the configuration bits.

Figure 1 also lists each signal on the FPGA fabric port. Individual functional blocks can implement all or only a subset of these signals. In general, it is a synchronous parallel-memory port, with separate read and write buses similar to the block RAM interface. Bus bits are numbered from least significant to most significant, starting at 0. All signals are active high.

Synchronous timing for the port is provided by the DCLK input, and all the other input signals are registered in the functional block on the rising edge of DCLK. Input (write) data to the functional blocks is presented simultaneously

with the write address and DWE and DEN signals before the next positive edge of DCLK.

The port asserts DRDY for one clock cycle when it is ready to accept more data. The timing requirements relative to DCLK for all the other signals are the same. The output data is not registered in the functional blocks. Output (read) data is available after some cycles following the cycle that DEN and DADDR are asserted. The availability of output data is indicated by the assertion of DRDY.

DRP Implementation in DCMs and MGTs

As mentioned earlier, the DRP is available in two major Virtex-4 functional blocks. Writing a specific value to a specific address will manipulate configuration bits and alter functions or attributes on the fly.

software tools to show the additional DPR signals. For example, writing a 04h to address 50h will change the M value to 5.

In the MGTs, the DRP allows advanced users to manipulate many attributes of the physical media attachment (PMA) and the physical coding sublayer (PCS). The new signals are part of the regular MGT primitive and can be operated by the application. The MGT implementation makes a large number of settings available for you to change dynamically. Various comma detect, channel bonding, and other attributes can be manipulated.

Conclusion

The Virtex-4 dynamic reconfiguration port provides an easy-to-use, block RAM-style interface that empowers you to

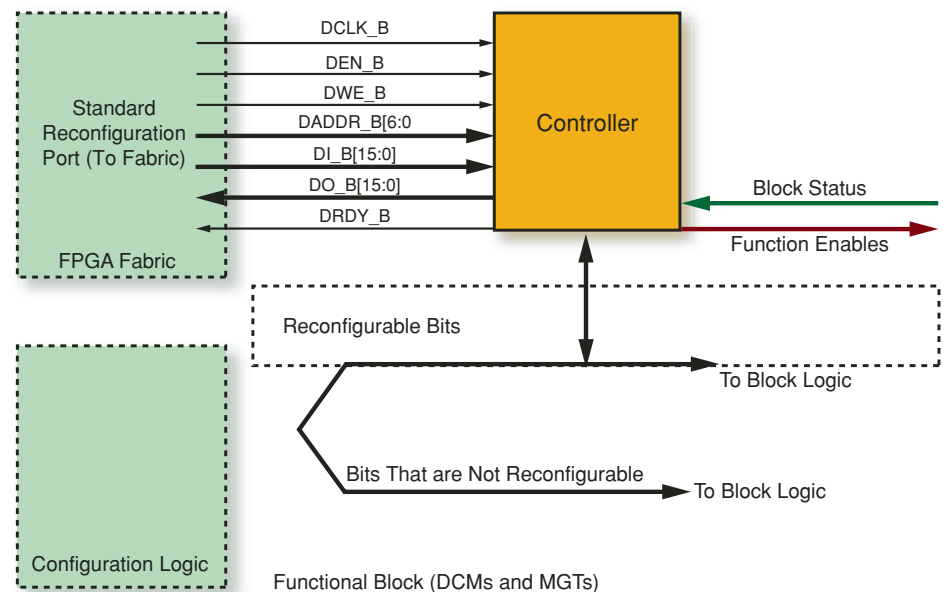


Figure 1 – Configuration changes

The user and configuration guides describe the address space (locations) and allowed values for each function.

In the DCM, the DRP allows you to make dynamic adjustments to the phase shift value of the digital phase shifter (DPS) and to change the multiply (M) and divide (D) values of the digital frequency synthesizer (DFS). A new primitive, DCM_ADV, has been added to the

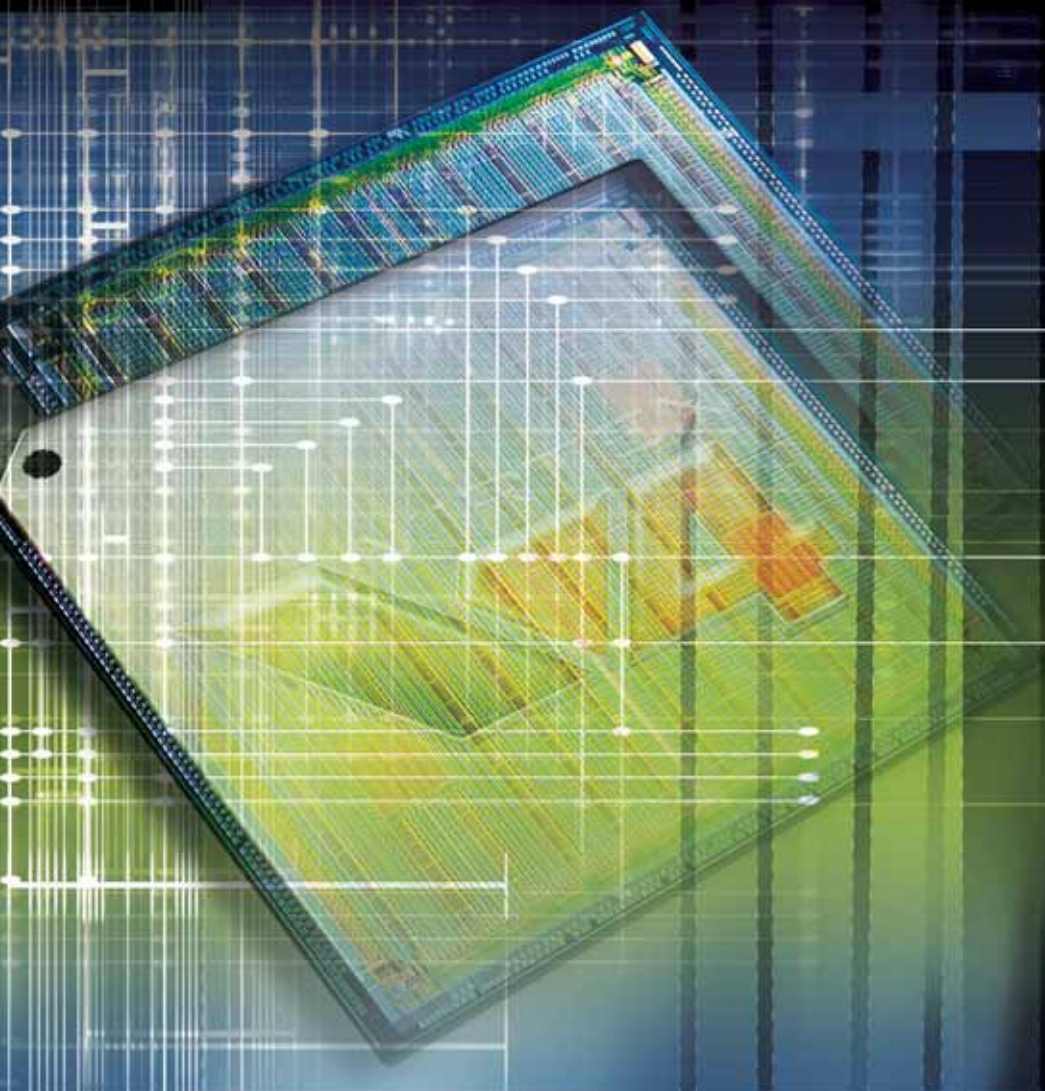
modify the functionality of your application while the device is operating. This leads to flexible implementations and an application that can adapt to changing conditions – without having to reconfigure an FPGA with a different bitstream from scratch.

For more information, see the configuration guide, www.xilinx.com/bvdocs/userguides/ug071.pdf.



Designing with the Virtex-4 Embedded Tri-Mode Ethernet MAC

Integrate the versatile Virtex-4 10/100/1000 Ethernet MAC into your next programmable SoC design.



by Hamish Fallside

Senior Manager, Systems Engineering,
Advanced Product Division

Xilinx, Inc.

hamish.fallside@xilinx.com

Ethernet is the predominant wired connectivity standard. The range of standard products for Ethernet is large, and it just got bigger with the introduction of the Xilinx® Virtex-4™ FX device family. Combining embedded Ethernet connectivity with the unique flexibility of the Virtex-4 feature set, Xilinx has created a compelling single-chip platform for solutions not possible with existing off-the-shelf products.

The Virtex-4 FX device family contains paired embedded Ethernet media access controllers (MAC) that are independently configurable to meet all common Ethernet system connectivity needs. Each Virtex-4 FX device contains either two or four MAC, implemented using Xilinx IP immersion technology, as shown in Figure 1.

Using standard Xilinx design products, you now have the unprecedented capability to create a huge range of customized packet processing and network end-point products for 10/100/1000 Mbps Ethernet.

An external physical layer device (PHY) is required for the MAC to connect to a network. The Virtex-4 FX device directly supports all standard serial and parallel PHY interfaces for both copper and optical Ethernet connections. In addition, Virtex-4 embedded RocketIO multi-gigabit transceivers can be used to drive Ethernet directly across PCB traces, such as serial backplanes, for in-system connectivity. PHY connections can be routed to any user pin or RocketIO block in the device.

In this article, we'll review the feature set of the embedded Ethernet MAC blocks in Virtex-4 FX devices, and offer some pointers on how you can start right away using them with standard Xilinx design tools, LogiCORE™ IP, and development boards.

Feature Set

The Virtex-4 Ethernet MAC addresses all common configuration requirements for embedded Ethernet connectivity, and is fully compliant to the IEEE802.3-2002

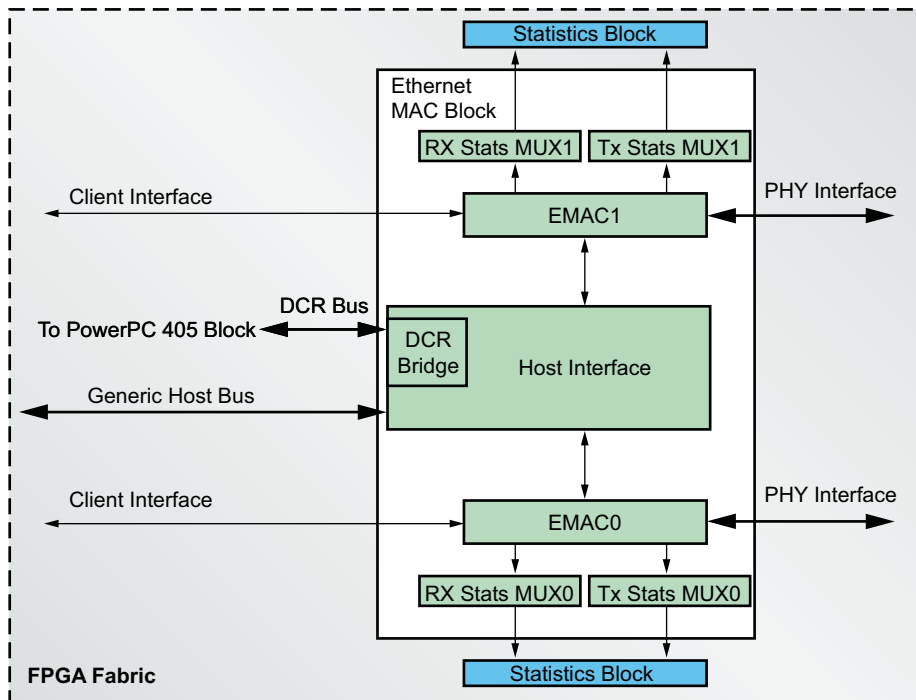


Figure 1 – Embedded Virtex-4 Ethernet MAC Block, with interfaces to FPGA resources

specification. It will allow you to build Ethernet systems that support VLAN, jumbo frames, and end-to-end flow control.

Built-in hardware address filtering reduces the burden on software of processing unneeded frames. You can independently configure each MAC for multiple rates and topologies:

- 10 Mbps or 100 Mbps full- and half-duplex
- 10/100 Mbps full- and half-duplex
- 1000 Mbps full-duplex
- 10/100/1000 Mbps full-duplex

When used in multi-rate modes, auto-negotiation support is provided.

Connecting the MAC to external PHY and optical modules is supported through the PHY interface to the FPGA fabric. This provides flexible use models for the MAC, allowing, for example, attachment to a shared processor bus or to custom packet processing hardware.

Controlling the MAC in your system is performed through the host interface, which provides flexible software access to the internal registers. Each MAC pair shares a common host interface, which can be

directly accessed by the embedded PowerPC™ 405 device control register (DCR) bus, or from the FPGA fabric.

Let's describe each of these interfaces in more detail.

PHY Interfaces

Your application will require connection to a particular medium – copper, fiber optics, or one of your own invention. The PHY interface provides many options to meet your requirements.

All common interfaces to external media are directly supported in the PHY interface. As the PHY interface is routed to the outside world through FPGA fabric, creating “bump-in-the-wire” solutions in FPGA fabric is straightforward.

PHY interfaces fall into two categories: one using SelectIO™ resources and another using RocketIO serial transceivers.

The first category is typically used to connect to a discrete external PHY:

- Media independent interface (MII) for 10/100 Mbps
- Gigabit MII (GMII), and reduced GMII (RGMII) for 10/100/1000 Mbps

The second category will also connect

directly to a discrete external PHY, and is commonly used to connect to small form-factor pluggable (SFP) modules for both optical and copper connectivity:

- Serial GMII (SGMII) for 10/100/1000 Mbps
- 1000BaseX for 1000 Mbps

These interface options have 9-bit signaling that connect to the RocketIO. Embedded state machines in the MAC provide University of New Hampshire-certified operation for link initialization using these options.

A MII management (MIIM) interface is also included, which allows your software to access external PHY registers through this standard IEEE interface. The registers are accessed via the address map in the host interface.

Host Interface

For your software to control the MAC, a host interface provides access to the internal registers. A dedicated DCR bus connects the embedded PowerPC directly to the host interface, requiring no additional FPGA resources. Alternatively, the host interface can also be accessed directly from the fabric, providing a flexible solution for porting legacy driver software. Each pair of MAC shares a single host interface.

The registers accessed through the host interface are used by driver software to initialize and control the MAC during operation. All register values may be preset at power-on from the FPGA fabric. This allows the MAC to be used by applications that do not include a processor and software. The registers provide access to the following settings:

- Independent receiver settings for reset and enable, pause frame address, jumbo and VLAN frame enables, half/full duplex, and passing frame check sequence (FCS) to the client
- Independent transmitter settings for reset and enable, inter-frame gap (IFG) adjustment, jumbo and VLAN frame enables, half/full duplex, and FCS from client

- Independent flow control enables for receiver and transmitter
- RGMII/SGMII status, and speed for fixed and negotiated settings
- Management interface enable and clock rate
- Receive-side address filter access – uni-cast and multi-cast address entries

The address filter provides a single uni-cast and as many as four multi-cast Ethernet addresses that are used to match against the destination address of incoming frames. You can set the filter to optionally discard incoming frames that do not match the stored addresses or to simply flag when a match occurs, allowing you to make routing decisions for received frames at hardware speed rather than in software.

Client Interface

Ethernet frames are passed between the MAC and your design across the client interface, which is divided into receive and transmit sides.

Receiver Side Client Interface

On the receive interface, frame errors and unmatched frames are signaled to the user logic. When flow control is enabled, any valid pause frames received will be flagged as invalid.

Transmitter Side Client Interface

The transmit interface will indicate collisions on half-duplex connections, and will corrupt a truncated frame in the case of FIFO starvation in the middle of a frame. When flow control is enabled, the transmitter interface will automatically assert back pressure on the client when a pause request frame is received from the remote host.

Flow Control and Statistics Vectors

A separate flow control interface allows the client to make pause requests to the far end, allowing the pause interval to be set for each individual request. Separate interfaces provide separate statistics vectors for the receiver and transmitter portions of the MAC. The IEEE-defined statistics are updated on a per-frame

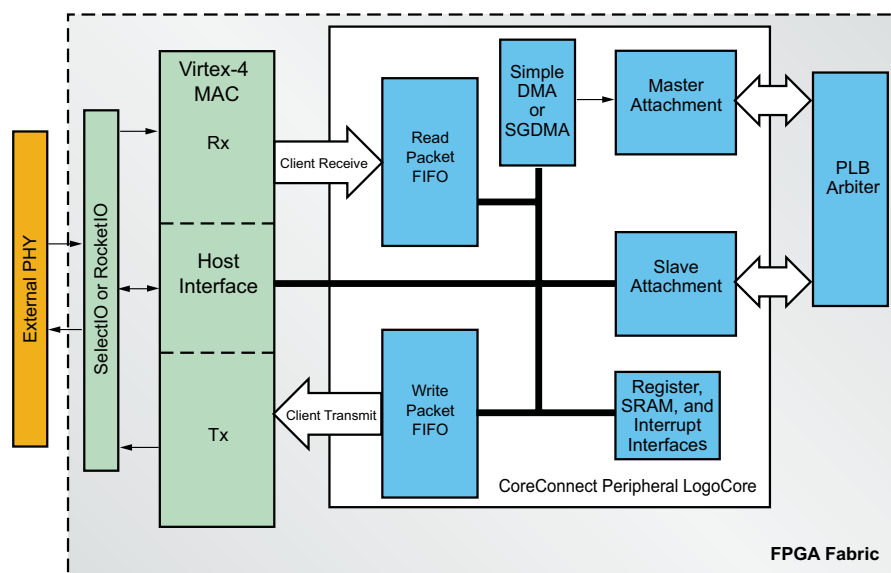


Figure 2 – Embedded MAC connected to embedded PowerPC as a PLB peripheral, with the addition of Xilinx CoreConnect LogiCORE IP

basis, and can be accumulated using circuitry in the FPGA fabric.

Over-Speed Operation

This feature allows you to clock the MAC at higher rates than allowed by the standard. The double-width interface on the client side means that your design can process frames at the same system frequency as normal operation, but at twice the data width, providing up to 2 Gbps in each direction.

Virtex-4 Ethernet MAC Use Models

The features described previously provide the Virtex-4 Ethernet MAC with multiple use models. Some examples of these are given here, but this should not be considered a complete list.

- Attach the MAC to CoreConnect PLB or OPB peripheral interface in FPGA fabric to embedded PowerPC or MicroBlaze™ processors, as in Figure 2.
- Create a custom interface to packet processing hardware implemented in FPGA fabric, such as protocol offload, DMA engines, embedded FIFO, and embedded block RAM. Figure 3 shows an example scheme for a Transmission Control Protocol (TCP) offload engine (TOE), and/or Remote Direct Memory Access (RDMA), as covered

by the iWARP protocols from the RDMA Consortium.

- Directly connect multiple MAC blocks to Virtex-4 embedded FIFO and external QDR and DDR memory for classification, policing, and switching applications, see Figure 3.
- Provide independent packet monitoring and statistics collection, using custom hardware in FPGA fabric that connects directly to the statistics interface of the MAC blocks.

Any of these use models may be connected to external PHY in multiple system topologies:

- Optical gigabit Ethernet connectivity – connect directly to external optical modules through the Virtex-4 RocketIO transceiver for 1000BaseX operation (Figure 4)
- 10/100 Ethernet connected to external copper PHY through RMII interface implemented between the MII PHY interface and SelectIO pins
- 10/100/1000 tri-mode Ethernet to external PHY or SFP module through SGMII connection to RocketIO transceiver, utilizing a RocketIO block

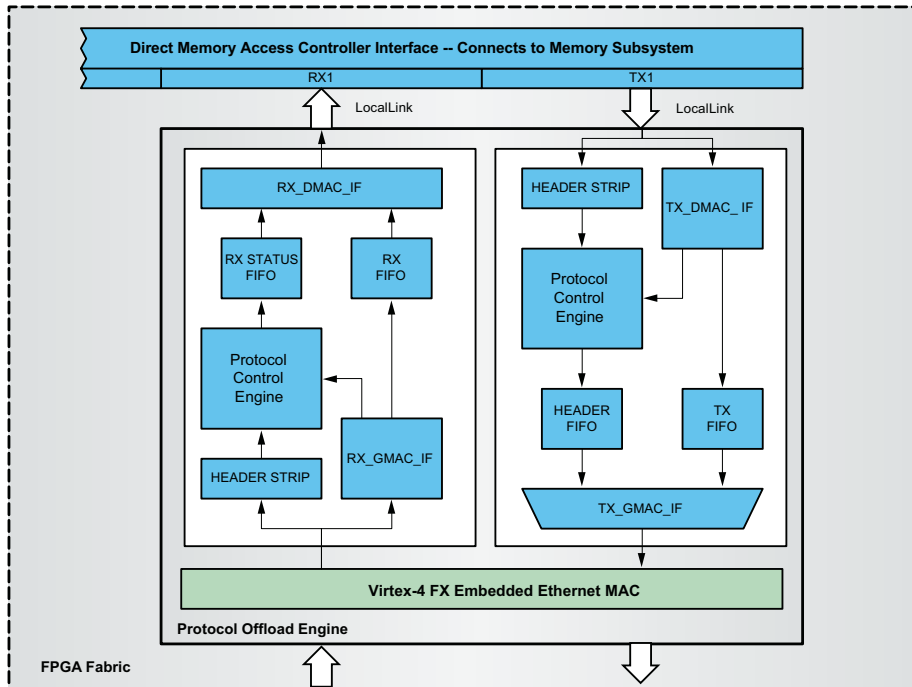


Figure 3 – Packet-processing end-point in Virtex-4 FPGAs using embedded MAC with additional logic for checksum offload, TCP segmentation offload (TSO), network address translation, and other standard or custom applications

Tools, IP, and Development Boards

Xilinx provides support for the MAC with tools, LogiCORE IP, reference designs, and Virtex-4 development boards.

Virtex-4 Embedded EMAC Wrappers

Available from the Xilinx CORE Generator™ tool, you can automatically generate HDL wrappers for the MAC instantiations in your design and completely configure the MAC through the GUI. A low-level software driver for the embedded PowerPC to access the MAC across the dedicated DCR interface will also be automatically generated.

Embedded Developers Kit (EDK)

The EDK tool enables you to build a complete processor subsystem around the MAC. The tool includes standard Xilinx LogiCORE IP to connect the MAC as a CoreConnect peripheral, and will automatically generate a software driver.

Xilinx Ethernet LogiCORE IP and Reference Designs

Much of the legacy Virtex-II Pro™ Ethernet collateral will be reusable with the Virtex-4 MAC.

Reference designs are available that demonstrate useful techniques for opti-

mizing your Ethernet system designs. The LocalLink LLTEMAC checksum offload peripheral, available with the Gigabit System Reference Design (XAPP536) demonstrates how to accelerate the TCP performance of your network endpoint.

Development Boards

Xilinx provides a family of development boards for immediate prototyping of your system design. These include:

- The ML403, a low-cost development platform featuring the Virtex-4 FX12 device, includes a tri-speed Ethernet PHY for Ethernet copper connectivity
- The ML405 development board provides a superset of the ML403, with additional serial connectivity options enabled by the Virtex-4 FX20 RocketIO transceivers

All Xilinx and partner-developed boards are available from the “Xilinx on Board” section of the Xilinx website.

Conclusion

The embedded tri-mode Ethernet MAC in Virtex-4 FX devices provides unparalleled flexibility for today’s Ethernet systems designers; spanning:

- Hub, switch, and router systems topologies
- Tightly coupled network processing functionality utilizing embedded processors and custom logic
- Embedded processing shared bus subsystems
- Direct low latency connectivity to packet storage
- Cost effective interoperability with future, current, and legacy physical layer standards

In short, the Virtex-4 FX family enables you to customize your solution for the Ethernet topology and feature set that your application requires. To find out more, please follow the Virtex-4 links on the Xilinx website, www.xilinx.com/virtex4/.

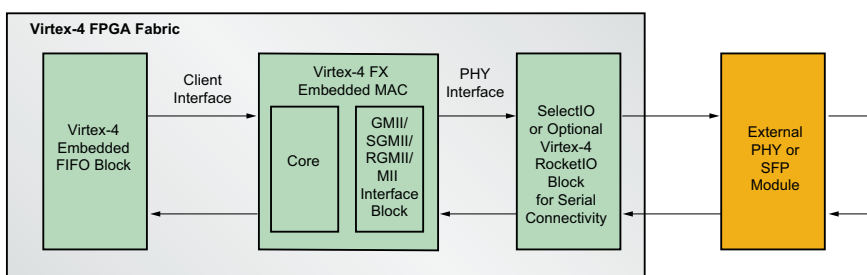


Figure 4 – Multiple Gigabit Ethernet MAC in a switch/router configuration; Virtex-4 embedded FIFO blocks provide intermediate packet storage in the fabric.

Emerging Design Methodologies Elicit the Power of Virtex-4 FPGAs

Adopt a broader design flow methodology instead of the traditional point-tool approach.



by Darren Zacher
 Technical Marketing Engineer
 Mentor Graphics Corporation, Design Creation and
 Synthesis Division
darren_zacher@mentor.com

Customers in today's demanding communications and consumer applications need to attain unprecedented levels of capacity and performance while reducing power consumption and overall cost. With the introduction of high-end devices into the marketplace, more of these applications are being addressed by FPGA solutions.

As professional programmable logic designers, you are always searching for better ways to create value and differentiate your products. To do so effectively, you need to adopt comprehensive, high-productivity design flows instead of point tools to crack new design challenges and take advantage of the benefits of the latest programmable silicon platforms.

Multiple Platforms, Unprecedented Opportunity

With the release of Xilinx® Virtex-4™ devices, you can enjoy twice the density, twice the performance, and half the power consumption of previous Xilinx FPGA families. If you seek sheer DSP performance, you might prefer Virtex-4 SX FPGAs, which offer 256 GigaMAC/s performance

for 18-bit operations. The LX family of FPGAs offers higher performance logic; with FX devices, you can explore embedded processing and high-speed serial connectivity applications. These three platforms, comprising a complete selection of 17 devices, collectively offer a compelling alternative to ASICs and ASSPs.

To fully exploit this immense potential, design teams must consider moving away from serial, iterative, point-tool approaches that involve designing or re-designing from scratch. To manage non-recurring engineering time and costs and create efficient, reliable flows, you must clearly identify which of the various "building blocks" you need to focus on when using a platform approach to successfully implement a high-end design.

Typical building blocks may include:

- Intellectual property such as internal company, Xilinx, or third-party IP
- Lower-level blocks used in the context of a bottom-up design flow
- Algorithms via C or C++ or MATLAB™
- RTL blocks
- Embedded processors
- I/O interfaces

By using a comprehensive, methodical design flow, you can effectively optimize these blocks in a multimillion-gate device.

As high-end FPGAs approach ASIC-level performance, designers are adapting many advanced ASIC techniques for FPGA design. The complex FPGA design flow shares some commonality with ASIC design; for instance, RTL simulation remains basically unchanged. But certain subtle differences exist under the hood, and many steps are fundamentally different. The pre-built nature of FPGAs implies a "use or lose" approach to features or capabilities, so you must match functional requirements with the device architecture. Thus, common steps such as synthesis or place and route all differ subtly in the FPGA domain.

You can use C++ synthesis techniques borrowed from ASIC flows to target FPGAs. C++ specifications are much less tied to any specific hardware than the corresponding RTL code.

Another technique, physical synthesis, illustrates the subtleties involved when the same general approach is used for both ASICs and FPGAs. Physical synthesis requires a detailed understanding of the FPGA's hardware structure. At the very least, physical synthesis tools must be more specifically targeted to FPGA architectures.



A typical high-end FPGA design flow should encompass such tasks as:

- Early design rule checking
- Higher level design abstraction
- Functional and system-level simulation and verification
- Advanced physical synthesis techniques

Let's describe each of these in more detail.

Integrated Approach to Design Creation

In terms of design entry, the need to create faster, larger, and complex designs packed into the latest FPGA devices within the shortest possible time presents significant challenges. The high availability of configurable logic in platform FPGAs that include hard ASIC macros – such as embedded processor blocks and complex I/O standards – has truly enabled programable SoC, where a serialized design approach would not work. Only a system-level RTL design concept, used in parallel with multiple aspects of managing and optimizing the high-level design creation process, will ensure success.

Large design projects mandate the collaboration of several engineers or engineering teams, often belonging to separate companies and typically distributed in different geographic locations worldwide. This team-based approach raises the importance of a consistent design coding style for teams to share code effectively.

Teams invariably comprise experienced project leaders and designers alongside less experienced junior engineers working on the various building blocks of a design. The resulting skill diversity makes the need for consistency critical. It is imperative that companies carefully scrutinize the planning and creation process to identify poor design styles, incorrect design rules, and syntax/semantic errors at the earliest possible stage before even attempting to tie the building blocks together or simulate/synthesize the design.

In bigger designs, it is not unusual for multidisciplinary design teams to focus on and optimize only a portion of the device. As the system is defined in RTL by combining both vendor and internal IP (and for

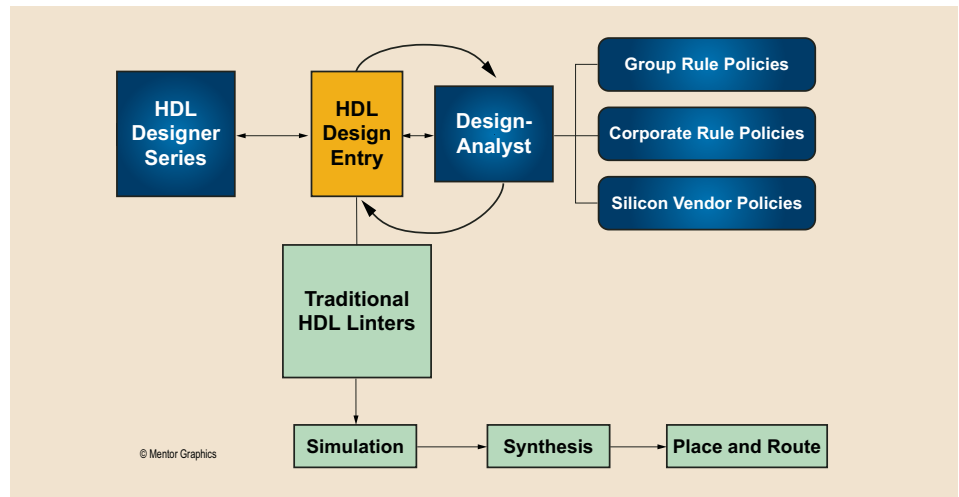


Figure 1 – When used in tandem for concurrent design entry and checking, interactive HDL visualization and creation tools can increase design quality, reduce iterations, shorten simulation and synthesis cycles, and improve testability and reuse in high-end FPGAs.

those applications utilizing DSP functionality, RTL generated algorithmically), you will need an integrated system design approach to help synchronize the development of each specific part of a large, high-capacity FPGA.

From the configuration of the embedded processor to logic development and high-speed I/O assignment, the ideal synchronization of these teams and processes is required to deliver an optimized field-programmable SoC. The merging and management of these multiple disciplines to generate the system-level RTL and associated design files is a huge task best handled by a comprehensive and flexible environment.

To reduce development cost and time to market, 80-90% of projects may now include both re-work of an existing design as well as reuse of previously designed components or IP, whether internal or purchased. Because this trend is expected to increase, you need to ensure that your components/subsystems are designed to be reusable and conform to established design reuse rules.

Through cooperative efforts in the design community and internal corporate standardization, the industry has developed a number of reuse methodology guidelines that can be checked using automated tools. Tools such as Mentor Graphics® HDL Designer Series™ (HDS) can help design teams successfully integrate both hard and soft IP (such as PowerPC™ and MicroBlaze™ processors).

Larger designs at higher speeds have prolonged traditional simulation cycles.

Similarly, synthesis can become a protracted, iterative process in order to achieve desired performance goals. You need to maximize the productivity of potentially long EDA tool runs by ensuring that as many code errors as possible are found and fixed before the start of simulation and synthesis (Figure 1).

Equally important are integrated connections to advanced tools such as DesignAnalyst™ and Precision® Synthesis from Mentor Graphics to ensure against errors and reduce iterations, as well as integration with any third-party EDA tools through a flexible integration mechanism. Through static design checking or “linting” products, you can perform many different forms of checking during the design creation process.

Interactive HDL visualization and creation tools provide automatic documentation features and reporting as well as intelligent debug and analysis to effectively manage FPGA designs. Moreover, tight bi-directional communications with PCB tools from within the design creation process shorten design cycles by integrating and synchronizing HDL design with PCB design, eliminating time-consuming manual steps.

Higher Abstraction Levels Speed Hardware Design

For the first time, professional design engineers are literally struggling to keep pace with Moore's Law, which makes it difficult to fully utilize the capacity of 90 nm ASICs

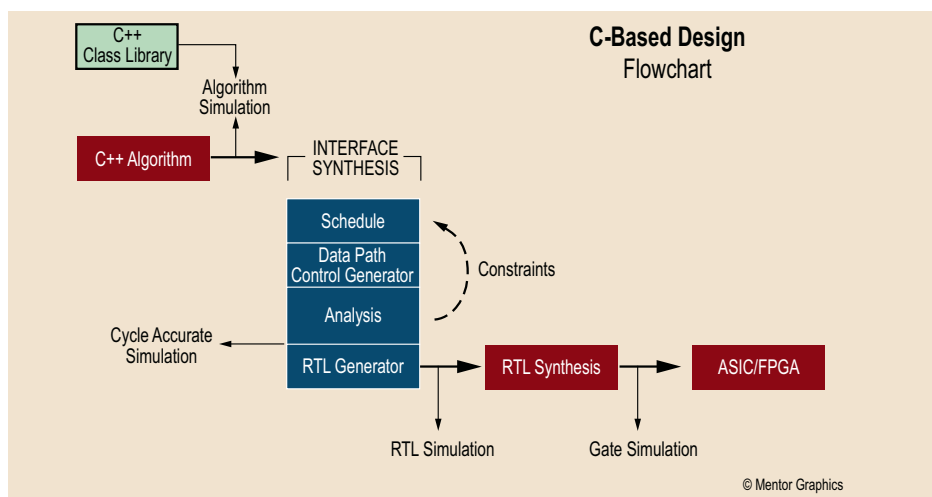


Figure 2 – An optimized C-to-RTL synthesis flow promotes a higher level of design abstraction and gives you the flexibility to easily transition from one implementation technology to another.

or efficiently target the complex structures found in domain-specific FPGAs. Algorithmic C synthesis (Figure 2) promises to raise the abstraction of hardware design by providing a new, more abstract entry point, benefiting both ASIC and FPGA hardware designers. But to understand the need for higher abstraction languages, you must first analyze the problems with existing RTL methodologies.

The design complexity of new DSP applications has outpaced traditional RTL capabilities. To create hardware implementations for blocks of computationally intensive algorithms using RTL, design teams must iterate through several steps, including micro-architecture definition, handwritten RTL, and area/speed optimization through RTL synthesis. This manual process is slow and error-prone. In the final result, both the micro-architecture and technology characteristics become hard-coded into the RTL description. This hard coding renders the whole notion of RTL reuse or retargeting impractical in real applications.

An optimized C-to-RTL synthesis flow not only promotes a higher level of abstraction, it also gives the design team the flexibility to transition from one implementation technology to another. You can tune the hardware for high-performance parallel implementations or smaller, more serial implementations.

Using this approach to describe functional intent (offered in the Mentor

Graphics Catapult™ C Synthesis tool), you can move up to a far more productive abstraction level for designing hardware. As hardware designers, you can reduce implementation efforts by as much as 20X while creating a more repeatable and reliable design flow.

The ability to select fundamentally superior micro-architectural alternatives allows you to create designs of better quality than traditional RTL methods. Finally, this approach closes the conceptual gap between algorithm designers modeling in C/C++ and hardware designers working at the RTL abstraction level.

Simulation and Verification Challenges

Using standard RTL verification methods in high-capacity FPGAs quickly diminishes the benefits of faster hardware creation. The current execution speeds of software validation platforms and RTL verification environments are insufficient to quickly test design functionality. Design verification takes significantly longer than design development because of the limited speed of RTL simulators and the time needed to manually create an RTL test bench.

Additionally, C/C++ simulation (although upwards of 10,000X faster than RTL) may be inadequate to validate the original algorithm given the data-intensive nature of DSP designs. These challenges are in fact opportunities for both algorithm development and system validation through the use of accelerated simulation.

High-level design verification flows are now turning to address rapid algorithm validation and verification, using hardware acceleration by leveraging the benefits of a SystemC verification environment. These flows begin with the algorithm designer validating designs in C++ and end with the hardware designer verifying the algorithm in RTL.

This method of using high-level C/C++ synthesis in combination with a SystemC verification environment provides an automated path from algorithm development to synthesized RTL running in an FPGA prototyping environment. Executing the algorithm directly in hardware gives algorithm designers the ability to validate algorithms and hardware designers the ability to validate the entire system at or near real-time speeds.

The use of SystemC as a verification environment permits both algorithm and hardware designers to use the same test bench and test vectors, eliminating the need for manual test bench creation. The combined approach of hardware acceleration of C/C++ algorithms in a SystemC verification environment provides a push-button solution for accelerated algorithm development and system validation.

Balancing the Cost/Timing Closure Equation

An essential step in realizing a high-capacity FPGA design is to optimize that design for both timing and cost. Timing closure challenges are well known. Using stand-alone logic synthesis with place and route can be non-deterministic by nature, especially for large devices.

Designers tend to write and rewrite RTL code and constraints to try and coax the place and route tool to do their bidding. Once you go down this path, you then must iterate through place and route – the most time-consuming step in FPGA design – before gaining any visibility as to whether your changes were a step in the right direction or if they only served to further exacerbate the problem.

Similar to optimization for timing, the process of achieving true “cost closure” involves a reduction in area to reduce FPGA part cost, or a reduction in the total

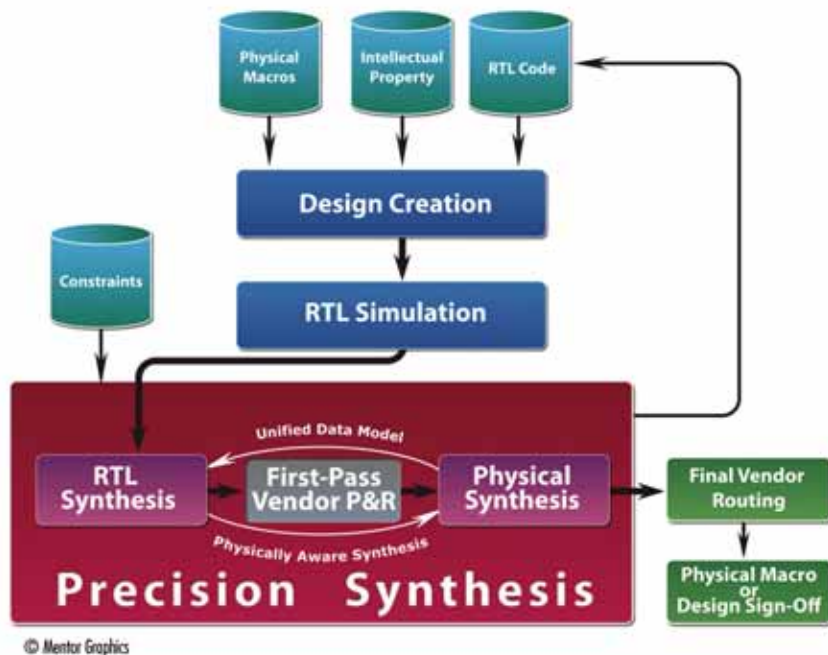


Figure 3 – High-end FPGA synthesis tools should ideally consider FPGA vendor placement results up-front, and only then begin to manipulate the design using physical synthesis – integrated with logic synthesis in a unified data model – to converge on timing.

cost of the design by increasing levels of abstraction and design reuse. The irony is that once you attain a successful implementation, any change – no matter how small – in the design or architecture threatens to obsolete that success. This unpredictability negates the reduced cost and time-to-market benefits of using programmable logic in the first place.

Increasing die sizes place additional burdens on the extant methodologies. A large die poses a significant challenge in obtaining repeatable, high-quality placements out of current placement algorithms. The larger die size is now widening the distribution curve of net delays grouped by fanout, the basis behind industry-accepted wire delay models.

This widened distribution has a degrading effect on the accuracy of fanout-based wire delay models. In larger devices, interconnect delay dominates performance for FPGA platforms. Because fanout-based delay estimates in FPGAs struggle to model even a simplified version of physical reality today, you can see why optimization decisions based on a wire-load estimate are often ineffective. Worse, physical proximity cannot always relate directly to delay, so tradi-

tional floorplanning falls painfully short. Advanced physical synthesis techniques can solve these issues in several ways.

First, to improve accuracy and reduce design iterations, you must consider real interconnect delay and physical effects up front (Figure 3); combining logic and physical synthesis is critical for the design of larger, high-performance FPGAs. Some physical synthesis alternatives available today are based solely on technology borrowed from the ASIC implementation space.

In reality, forcing an ASIC methodology – and mentality – on the FPGA world cannot work. Such approaches essentially try to outsmart the vendor placement and may show promise in certain situations, but most cannot match the performance of a tool that leverages the FPGA vendor's post-layout information to provide accurate physically aware synthesis.

Second, FPGA-oriented physical synthesis solutions need to take into account successful implementation experience that you have previously developed. For instance, when you complete a modular design and have optimized performance for a portion of it using physical synthesis, a

good tool must ensure that you can take full advantage of these optimizations and reuse them on subsequent designs.

Physical synthesis in FPGAs is growing beyond the ASIC model to be a valuable part of cost minimization and component reuse strategies. When investing in a synthesis tool with a highly deterministic process for improved results, look for technologies and algorithms that not only optimize designs for cost and timing, but also enable you to translate your professional experience and previous design implementations at the physical level into faster time to market in subsequent designs.

Any tool used in professional FPGA design (including the Precision Synthesis tool from Mentor Graphics) should consider FPGA vendor placement results as soon as possible, and only then begin to manipulate the design using physical synthesis – integrated with logic synthesis in a unified data model – to converge on timing at a lower cost.

From Point Tools to ESL Design Flows

Every designer stands poised to benefit from the new standard set by Virtex-4 high-performance FPGAs. The next-generation challenge faced by mainstream FPGA EDA tool vendors is to leverage point-tool expertise and thus meld apparently contradictory trends – higher levels of abstraction on the one hand and greater dependence on specific physical characteristics on the other – into a coherent design methodology and highly productive flow.

In keeping with these advances, EDA tool companies will continue to extend and improve their comprehensive, integrated design flows spanning all levels of abstraction. Mentor Graphics continues to be a technology leader in this space. Designers must take advantage of EDA tools that now address both physical and electronic system-level (ESL) challenges of high-end FPGAs, and thus realize the unprecedented potential of these devices as ASIC replacements in new SoC designs.

To access the latest product news, application notes, and case studies, evaluate new design flows, or schedule a product demonstration, visit www.mentor.com/fpgal.

Integrating EDK-Created Embedded Processor Subsystems

by Andy Norton
President
Comm Logic Design, Inc.
andy@CommLogicDesign.com

The availability of embedded processor subsystems in FPGAs opens the door to a myriad of applications, including embedded network processors, flexible sandbox prototyping, control plane and data path subsystems, and exception handling processors. Today's FPGAs integrate existing IP cores, interfaces, custom processing engines, and now embedded processor subsystems. You can easily instantiate these subsystems into a top-level HDL design just as you would integrate off-the-shelf IP.

Xilinx® Virtex-4™ FX FPGAs integrate a higher performance IBM™ PowerPC™ core with the new Auxiliary Processor Unit interface. The direct connection to the FPGA fabric facilitates advanced coprocessor designs.

You can use Xilinx Platform Studio/EDK software to design embedded processor subsystems in FPGAs with embedded PowerPC hard processor cores or with Xilinx MicroBlaze™ soft processor cores. Although off-the-shelf peripheral cores and MicroBlaze soft cores are synthesized using XST during EDK platform generation, the overall FPGA project and custom peripheral cores are synthesized with Synplicity® Synplify Pro® 8.0, leveraging new features and superior quality of results.

EDK Subsystem Project Flow

All projects begin by defining an overall FPGA directory structure. The embedded subsystem should reside in its own sub-directory. For example:

```
fpga_project
/ doc          spec and documentation
/ src          RTL source code files
/ constraints  .ucf, .sdc files
/ sim          simulation files
/ syn          synthesis project files
/ pnr          place and route files
/ ppc_subsystem embedded processor subsystem
```

Creating a new EDK project in /ppc_subsystem results in a system.xmp project file. Next, EDK Project Options must indicate that it is a subsystem by setting:

1. Design Hierarchy to SubModule

Specifying the top-instance name of the embedded subsystem (ppc_subsystem). The indicated top-instance name will be used when instantiating the subsystem in the overall top-level design.

Use Synplify Pro as the primary synthesis tool for complex designs containing embedded processors.



2. Synthesis Tool to None

This indicates that no synthesis tool is used to synthesize the overall design within EDK (the instantiated subsystem will be included later in the Synplify Pro project), although EDK will have used XST (and possibly Synplify Pro) in the platform creation of the subsystem and its peripherals.

3. Implementation Tool Flow to ISE™

Although Synplify Pro supports mixed languages, you can select Verilog™ or VHDL for EDK output files in Project Options/HDL and Simulation.

Platform Generation

You can create the embedded processor subsystem by using either the Base System Builder wizard, the GUI selection of peripheral cores, or direct text editing of the microprocessor hardware specification (MHS) file.

Once the MHS file has been constructed, Generate Netlist invokes Platform Generation. PlatGen constructs the netlist, builds and interconnects indicated peripherals, runs DRC checking for errors and warnings, and generates output files.



The EDK Platform generated directories and files include:

ppc_subsystem	top-level instance of the subsystem
/hdl	
system_stub.[vhd v]	HDL subsystem with Xilinx I/O primitives inserted
system.[vhd v]	HDL subsystem without Xilinx I/O primitives
wrappers.[vhd v]	implementation netlist peripheral files with instantiated wrappers
/implementation	
system_stub.bmm	BMM file with top-level subsystem instance in path
system.bmm	BMM file without the top-level subsystem instance in path
peripherals.ngc files	XST-generated peripheral files

PlatGen will generate two top-level files in /hdl: system_stub.v and system.v. System_stub.v instantiates system.v and adds I/O insertion as Xilinx primitives for all top-level ports. With the processor as a subsystem, system_stub.v is not used because there are other cores, subsystems, and logic in the design. For example, clock signals could be generated by top-level instantiated DCMs and subsystem signals could go to other modules at the same level of hierarchy instead of off-chip.

Also, using Synplify Pro, the I/O insertion is automatic; you don't need to explicitly instantiate BUFG, IBUF, or OBUF primitives for most I/O standards.

Choosing to instantiate system_stub.v as our subsystem would then require editing, removing, or modifying the I/O insertion for the ports not directly connected to an external pin. Once modified, rerunning

PlatGen would overwrite this file once again. Another choice might be to rename system_stub.v after editing the file; the downside to this approach is that port/subsystem modifications would require you to recreate the modified/edited file.

A better approach is to instantiate system.v directly in the top-level HDL. Synplify will take care of the necessary I/O insertion where required or, for I/O standards requiring I/O primitive instantiation (for example, LVDS), this should be done directly in the top-level HDL file. System.v is always correct as generated by EDK PlatGen and never needs to be modified. The one additional step required is at the top level, in the case of tri-state signals.

For example, you can define the project top-level ports as:

```
module fpga_top
(
    inout      [31:0]  ddr_dq,
);
```

PlatGen will generate system.v (in /implementation), bringing out the tri-state signals as shown in the instantiated ppc_subsystem:

```
system ppc_subsystem (
    .
    .
    .ddr_dq_I      ( ddr_dq ),
    .ddr_dq_O      ( ddr_dq_o ),
    .ddr_dq_T      ( ddr_dq_t ),
    .
    .
);
```

The EDK-generated system_stub.v – the file we don't want to use – added the IOBUF insertion, as shown here for each bus signal:

```
IOBUF
iobuf_28 (
    .I ( ddr_dq_O[0] ),
    .IO ( ddr_dq[0] ),
    .O ( ddr_dq_I[0] ),
    .T ( ddr_dq_T[0] )
);
```

Because we want to be able to instantiate system.v directly into our top level, we

must also add the required HDL to control the bidirectional signals:

```
genvar i;
generate
    for(i=0; i<=31; i=i+1)
        begin: ddrtri
            assign ddr_dq[i] = ddr_dq_t[i]
                ? 1'bZ : ddr_dq_o[i];
        end
    endgenerate
```

Now EDK-generated subsystem Verilog files do not need to be modified – only instantiated. Bi-directional signals are handled correctly and I/O insertion is either handled automatically by Synplify or explicitly instantiated as Xilinx primitives when required.

Memory Generation

PlatGen will also generate the required memory initialization files for the specified block RAMs coupled with DSOCM, ISOCM (PowerPC only), LMB (MicroBlaze soft processor core only), OPB, and PLB block RAM controllers.

PlatGen will produce two BMM (block RAM memory map) files in the /implementation directory: system.bmm and system_stub.bmm. A BMM file will be used in the ISE flow to indicate the logical data space used by the embedded subsystem and organization of the block RAM memory. In the case of our subsystem, system_stub.bmm would be used, as it contains the complete hierarchical path (because we specified the top-level instance of our subsystem in the project options).

During the ISE bitgen phase of the flow, a system_stub_bd.bmm file will be created in the /implementation directory, indicating the physical location of the block RAMs.

Synplify Project Flow

While XPS/EDK generates the embedded processor subsystem (/implementation/system.v), once created the ppc_subsystem is instantiated exactly as any IP block by adding it to the overall Synplify synthesis project. Whether the underlying embedded processor subsystem used XST, Synplify, or both to create the peripherals and generate the subsystem is irrelevant to the overall Synplify synthesis project.

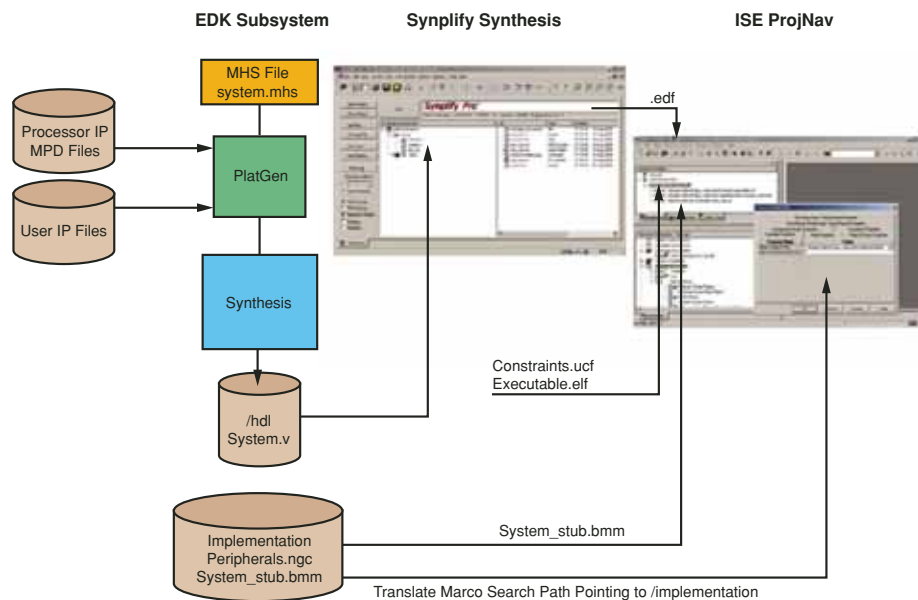


Figure 1 – Synthesis project design flow

A typical synthesis project flow, as shown in Figure 1, would follow this order:

1. Create a synthesis project
2. Add files to the synthesis project
project_top.v
/ppc_subsystem/hdl/system.v
(EDK-generated subsystem)
3. Synthesize and review the synthesized project
4. Use the generated output files in the ISE project
fpga_top.edf (top-level source file)
fpga_top.ncf (sdc-translated constraints file)

System.v contains the actual embedded subsystem with the peripheral wrappers instantiated. At the end of system.v are black_box definitions for each of the wrappers. Although Synplify doesn't recognize these XST synthesis directives, it does realize that it has to create black boxes and does so without modification.

Synplify will generate the warnings shown in Figure 2 because of the XST-generated synthesis directives and empty black box modules. Once reviewed and accounted for, these warnings can now be "hidden" using the Synplify Pro warnings

filter, as shown in Figure 3. The filter creates a project.prf file (Figure 4). This file can also be sourced in the Tcl window (source filename).

ProjNav ISE Flow

The /pnr directory is used for the Xilinx ProjNav ISE flow. The fpga_project.npl file is created by ProjNav indicating ISE project options.

[H]	[34]	CS141	Unrecognized synthesis directive attribute	system.v (444)	test_top.in (3)	17:10:10	Compiler Report
[E]	[17]	CG146	Creating black box for empty module ppc405_0_wrapper	system.v (1041)	test_top.in (24)	17:10:10	Compiler Report

Figure 2 – Synplify Pro 8.0 compiler warnings

Type	ID	Messages	Source Location	Log Location	Time	Report
[H]	MT108	Autocounter Mode is ON	-	test_top.in (117)	17:10:10	Mapper Report
[H]	FS164	The option to pack flops in the IOB has not been specified	-	test_top.in (129)	17:10:10	Mapper Report
[H]	MT130	This timing report estimates place and route data. Please look at the place and route data.	-	test_top.in (212)	17:10:10	Timing Report
[H]	MT137	Clock constraints cover only FF-to-FF paths associated with the clock.	-	test_top.in (214)	17:10:10	Timing Report
[E]	MT186	Black-box <leds_16bit_wrapper> is missing a user supplied timing model. This may cause timing errors.	system.v (1300)	test_top.in (183)	17:10:10	Mapper Report

Figure 3 – Synplify Pro 8.0 warnings filter

```
00001 # Message Filter
00002 log_filter -hide_matches
00003 log_filter -field type==Note -field id==MT186
00004 log_filter -field id==CS141 -field source_loc==system.v
00005 log_filter -field id==CG146 -field source_loc==system.v
00006 log_filter -field message==*translate*
00007 log_filter -field message=="Found counter*"
00008
```

Figure 4 – Synplify Pro .prf file

The following source files are added to the ISE project:

1. fpga_top.edf (Synplify top-level netlist with ppc_subsystem)
fpga_top.ncf (not added as an explicit source file; created from the Synplify constraints [.sdc])
2. /constraints/constraints.ucf (Xilinx constraints file)
3. /ppc_subsystem/implementation/system_stub.bmm

This file requires no modification, assuming that the subsystem instantiated in the top-level module uses the same instance name as generated by system_stub.v (that is, the top instance name indicated in the project options).

4. /ppc_subsystem/ppc405_0/code/executable.elf

An .elf file (pronounced "elf") is a binary data file that contains an executable CPU code image ready for running on a CPU. These files are produced by software compiler/linker tools. Data2BRAM uses .elf files as its basic data input form.



ISE Translate Properties must set the Macro Search Path to point to the /ppc_subsystem/implementation directory for it to find the .ngc peripherals that were black-boxed by Synplify, referenced in fpga_top.edf. These peripherals were created by XST during PlatGen.

Project implementation then follows a normal ProjNav flow producing translate, map, place and route, and timing reports.

You can easily incorporate embedded processor software changes made by the EDK GNU compiler into the final .bit file without hardware recompiles by running Generate Programming File, or alternatively, the Data2Mem utility. When using Data2Mem, the BMM file specified (-bm) must use the BitGen-generated system_stub_bd.bmm in the /implementation directory.

Custom Peripheral Cores

XPS provides a Create Peripheral Wizard that generates core description files and ensures that custom peripherals comply with the Xilinx implementation of the IBM CoreConnect PLB and OPB bus standard. The PLB and OPB buses will connect to an IPIF, allowing user logic to connect to the IPIC side of the interface. Unfortunately, the wizard currently supports only VHDL. Peripheral cores can also be created in Verilog, but cannot take advantage of the templates created by the wizard.

DCR and OCM bus IP cores are not currently supported through a template or wizard. DCR and OCM bus protocols are simple to understand, however, and you can easily create Pcores for these buses either in VHDL or Verilog. The current EDK-provided OCM buses now allow configurable multi-slave capabilities, providing an easy way to create low-latency slave-only peripherals.

You can integrate custom IP cores into the EDK project either as a black box

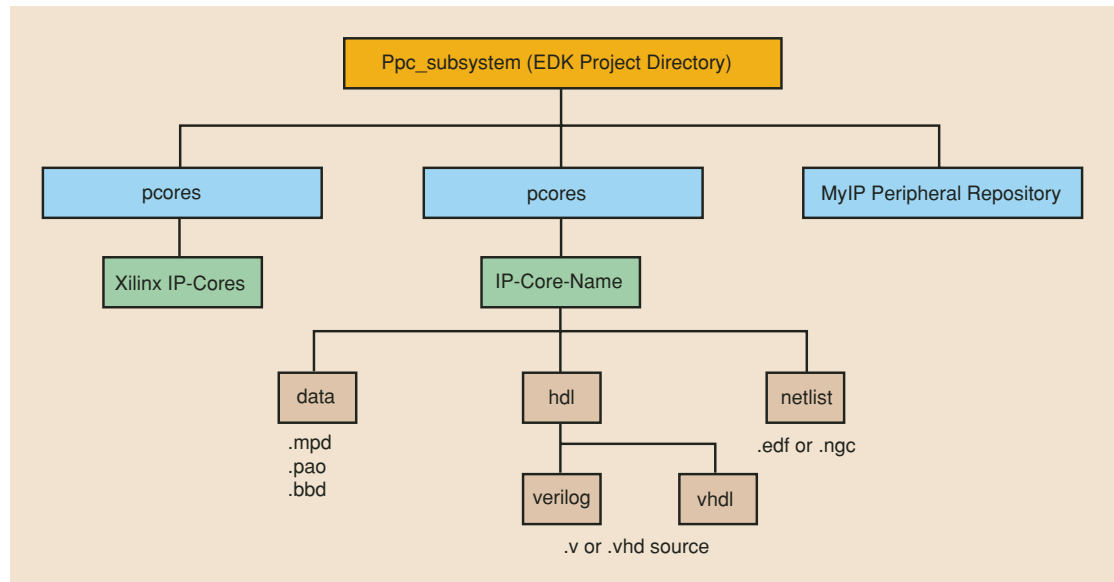


Figure 5 – Pcore directory structure

synthesized with Synplify or as an XST netlist. The Synplify-generated IP core requires associated MPD (microprocessor peripheral definition) and BBD (black box definition) files. The XST netlist is synthesized by PlatGen along with the system and requires MPD and PAO files.

Directory Structure

Figure 5 shows the required Pcore directory structure. PlatGen searches for IP according to the following priorities:

1. /pcores directory in the project directory
2. <library_path>/Library Name>/pcores if -lp option set (project options/peripheral repository)
3. \$EDK/hw/XilinxProcessorIPLib/pcores

Pcore Files

The Pcore HDL source files must be located in the /verilog or /vhdl directory if they are to be synthesized by XST with PlatGen. If the Pcore is provided as a Synplify-generated netlist, the EDIF must be located in the /netlist directory and indicate its black-box status in a BBD file. Required MPD, PAO, and BBD files for the peripheral must be placed in the /data directory.

The .mpd file specifies PORTs, PARAMETERS, BUS_INTERFACES, and OPTIONS. For Verilog files, the HDL option specified is OPTION HDL = VERILOG.

If XST is used as the synthesis tool for creation of the peripheral, the netlist option is OPTION IMP_NETLIST = TRUE.

If Synplify is used for the creation of the peripheral, the netlist option is OPTION IMP_NETLIST = FALSE. This would tell PlatGen to not run XST synthesis for this peripheral. A peripheral wrapper is still created and instantiated in system.v and the project synthesis run in Synplify would again create a black box for this peripheral.

Conclusion

You can easily integrate Xilinx embedded processor subsystems created using EDK into a Synplicity flow by instantiating the EDK-generated embedded subsystem into the top-level HDL design. You can use Synplicity tools not only as the overall project synthesis tool but also as the peripheral core synthesis tool in the creation of custom peripherals.

For more information, visit www.CommLogicDesign.com. Comm Logic Design is a Xilinx XPERTS partner focused on architecting, building, and delivering system solutions for wired-network, telecom, and storage applications.

Optimizing Virtex-4 High-Performance Designs

Synopsys Design Compiler FPGA can take your high-speed design to the next level of performance.

by Carlos Abraham
FPGA Synthesis CAE
Synopsys, Inc.
carlos.abraham@synopsys.com

Yanbing Li
Corporate Applications Engineering Manager
Synopsys, Inc.
yanbing.li@synopsys.com

Synopsys® Design Compiler® FPGA (DC FPGA) allows you to meet your high-performance design goals by using a powerful set of optimization algorithms and features specifically tuned for the Xilinx® Virtex-4™ architecture. These algorithms use special Virtex-4 resources such as the DSP48 block and block RAM to achieve the lowest overall area utilization and the optimal circuit timing performance.

Design Compiler FPGA Overview

Designs that target complex devices such as Virtex-4 FPGAs require the same power and flexibility in synthesis that only ASIC designers had access to in the past. DC FPGA is built on Design Compiler's industry-leading ASIC synthesis technology and then customized to include FPGA-specific optimizations to handle even the most challenging designs. FPGA-specific optimizations enable optimal mapping to FPGA basic primitives such as LUTs and complex components like RAM, multipliers, and DSP blocks.



DC FPGA includes innovative Adaptive Optimization™ (AO) technology to dynamically tune the synthesis algorithms based on the design context, as well as timing constraints to provide faster synthesis runtime and optimal timing. DC FPGA inherits Design Compiler's reliability – proven through the development of more than 125,000 ASIC designs. DC FPGA brings the powerful ASIC-strength synthesis of Design Compiler to FPGA designs.

In addition to AO technology, DC FPGA deploys a rich set of optimizations to achieve the best timing Quality of Results (QoR) for FPGA devices. These include:

- Constraint-driven synthesis and design space exploration
- Automatic finite state machine (FSM) extraction and optimization
- Automatic inference of special FPGA resources, such as RAM, ROM, multipliers, DSP blocks, shift registers, and global clock buffers
- Advanced datapath optimizations and module generation
- Logic and register duplication
- Register retiming and pipelining
- Critical path re-synthesis
- Across-boundary optimization
- Automatic gated-clock transformation

DC FPGA is part of a family of products from Synopsys that work in conjunction with the Xilinx ISE™ tool to streamline the FPGA design process.

In this article, we'll show how DC FPGA optimizes for high performance in Xilinx Virtex-4 FPGAs.

Constraint-Driven Synthesis

DC FPGA uses a true timing-driven synthesis engine. You can greatly influence the final implementation choice by specifying appropriate timing and design-specific constraints during synthesis. Therefore, we recommend that you drive DC FPGA synthesis with the same set of constraints as the Xilinx ISE tool.

At a minimum, you should specify appropriate design timing constraints such

as clock frequency, I/O offsets, and any timing exceptions applicable to your design (such as multicycle and false paths). Any other design-specific constraints – such as controlling special FPGA resource usage – could also be specified. For best performance, your design should not be over-constrained, which in some cases can lead to unnecessary increases in area.

Without any timing constraints, DC FPGA will perform area-based optimizations with good timing results. With proper timing constraints, DC FPGA applies the AO technology to explore the area-timing tradeoffs of various optimizations, selecting the final implementation that best fits your constraints.

For example, your timing goals enable DC FPGA to decide whether distributed RAM, block RAM, or a LUT with register-based implementation is sufficient for an inferred memory component in your design. Otherwise, DC FPGA optimizes for the lowest area utilization possible.

	Clock Constraint	Post-PAR Area (# of Slices)	Post-PAR Fmax (MHz)
Case 1	10 ns	105	260.1
Case 2	3 ns	116	334.8

Table 1 - Design example showing area-timing tradeoffs in DC FPGA

Table 1 shows two implementations for a small sub-module with two different clock constraints. The module is the critical one for a larger design of about 8,600 slices. The design contains a single clock domain with only one clock period constraint specified in DC FPGA.

In the first case, the module is constrained at 10 ns. DC FPGA exceeds the timing requirement after its area-based implementation and does not invoke the timing optimization phase. The critical path of the design runs through a series of carry logic.

In the second case, when a much tighter constraint (3 ns) is applied, DC FPGA performs aggressive timing optimizations and replaces the carry logic on its critical paths with parallel circuit structures built by LUTs. This results in a design with a slightly larger area but meets the new timing requirement,

which was impossible to achieve with the carry logic structure. At the overall design level, a 29% timing improvement is achieved with a minor area increase of 11 slices.

Flexible FSM Support

DC FPGA contains sophisticated FSM extraction and optimization algorithms to ensure optimum high-performance state logic implementation. Once the FSM is detected and extracted from the RTL code, DC FPGA's powerful state machine optimization engine performs various optimization schemes, such as optimizing unreachable states or removing duplicate states to produce the best logic implementation to meet timing.

At the same time, you have the flexibility to select a different FSM coding style such as one-hot, binary, gray, and zero-one-hot on a state-machine-by-state-machine basis, design basis, and global basis. This FSM encoding exploration flexibility allows you to customize the synthesis script to address design bottlenecks.

For an FPGA implementation, one-hot state implementations typically provide the best timing QoR for most designs at the expense of a higher register-to-LUT ratio. For most designs this is not a problem because of the register-rich architecture of FPGA devices.

High-Performance DSP Inference Capability

The availability of special FPGA resources such as block RAM, dedicated DSP slice, and carry logic combined with your specified design and timing constraints guides DC FPGA's specialized optimization algorithms to determine the best optimum circuit implementation.

DC FPGA is highly capable of inferring complex circuit topology from your design's RTL coding structure, effectively deciding the final implementation that best exploits the resources of the targeted FPGA. DC FPGA minimizes overall resource usage while providing the best circuit performance possible.

This powerful optimization feature allows DC FPGA to effectively infer and map complex logic configurations into special

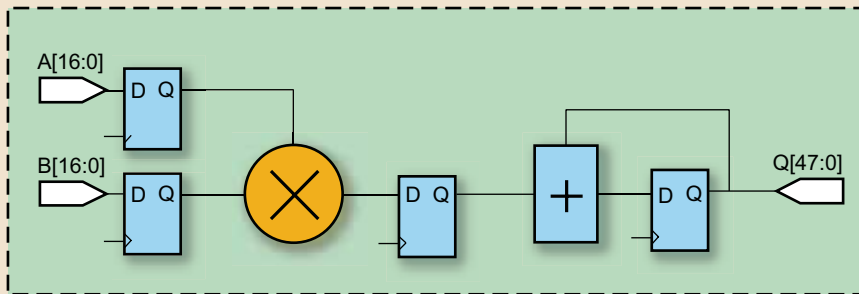


Figure 1 - Simple multiply accumulate (MAC) logic

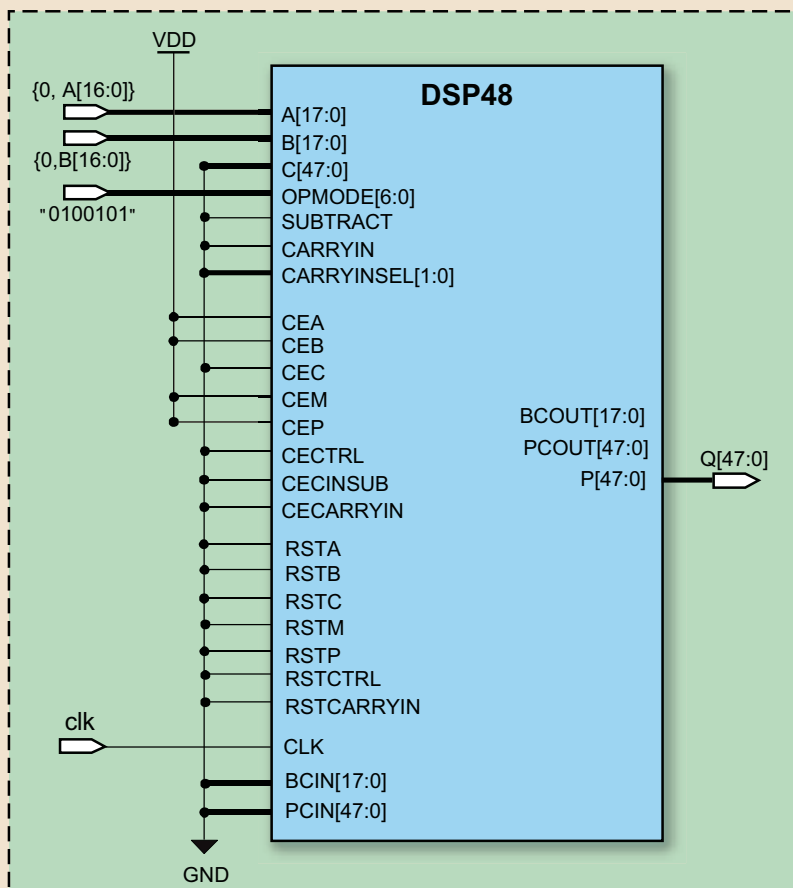


Figure 2 - DC FPGA single DSP48 implementation for MAC logic

resources such as the Virtex-4 dedicated DSP48 slice. To illustrate this powerful feature, Figure 1 shows a simple multiply accumulate (MAC) logic structure, where A- and B-registered input signals are multiplied. The registered multiplier intermediate output is then accumulated in the last adder stage, feeding the registered Q output signal.

The RTL code for this simple MAC function is:

```
module test ( Q, A, B, clk );
output [47:0] Q;
input [16:0] A, B;
input clk;

reg [47:0] Q;
reg [16:0] A_reg, B_reg;
reg [33:0] mult;

always @( posedge clk )
begin
    A_reg <= A;
    B_reg <= B;
    mult <= A_reg * B_reg;
    Q <= Q + mult;
end

endmodule
```

DC FPGA is able to effectively implement the logic configuration shown in Figure 1 in a single DSP48 slice, fully recognizing and taking advantage of the DSP48's embedded 18 x18 signed multipliers, accumulated adder mode, and integrated pipeline registers to obtain the highest performance system clock speed.

Figure 2 shows the final DC FPGA single DSP48 implementation without the use of other logic resources. The OPMODE control input pin of the DSP48 element is set to "0100101" to realize the overall MAC functionality mode intended by circuit topology, while the AREG, BREG, MREG, and PREG attributes are set to "1," respectively, to signify a single-stage register pipeline.

Furthermore, the high-performance DSP inference feature in DC FPGA supports very complex design topologies. Such topologies are extensively used in DSP-intensive applications such as a digi-

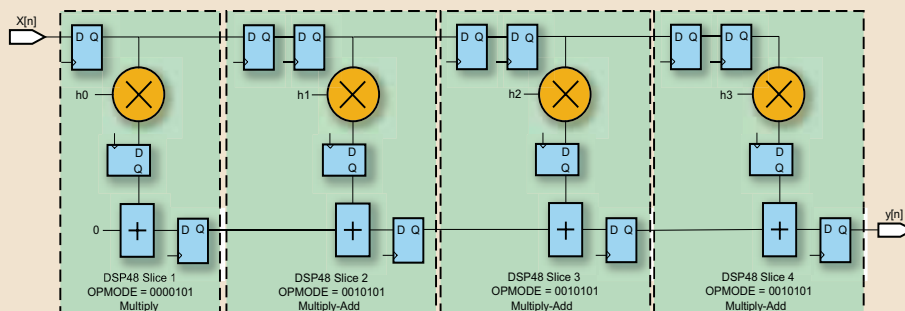


Figure 3 - Four-tap systolic FIR digital filter structures



tal FIR filter, commonly found in wireless communication applications.

Figure 3 shows the schematic of a four-tap systolic FIR digital filter structure. DC FPGA uses advanced DSP inference to implement this design in only four DSP48 slices without the use of external logic resources. The integrated pipeline registers are further exploited for faster clock throughput performance for this type of filter structure.

The following shows the RTL code for the systolic FIR filter:

```
module test ( Yn, Xn, h0, h1, h2, h3, clk );
output [47:0] Yn;
input  [15:0] Xn, h0, h1, h2, h3;
input      clk;

reg [15:0] X [7:1];
wire [15:0] h [3:0];
reg [32:0] mult [3:0];
reg [47:0] pcout [3:0];
wire [47:0] Yn;
integer i;

assign h[3] = h3, h[2] = h2, h[1] = h1,
h[0] = h0;

always @( posedge clk )
begin
X[1]  <= Xn;
mult[0] <= h[0] * X[1];
pcout[0] <= mult[0];

for (i=1; i <= 3; i=i+1)
begin: my_for_loop_block0
X[2*i] <= X[2*i-1];
X[2*i+1] <= X[2*i];
mult[i] <= h[i] * X[2*i+1];
pcout[i] <= pcout[i-1] + mult[i];
end //my_for_loop_block0
end

assign Yn = pcout[3];

endmodule
```

DC FPGA can also implement other complex logic configurations in a DSP48 slice. Table 2 shows a sample of some of these complex logic structures.

The designs shown in Table 2 were synthesized using DC FPGA and place

and routed using Xilinx ISE 6.3i Service Pack 2, while targeting an XC4VFX20-11 Virtex-4 device. The purpose of this exercise is to show the performance and area improvements performed by DC FPGA's advanced DSP inference capability. Each design was synthesized with and without DSP inference enabled during synthesis.

Conclusion

Complex devices such as Virtex-4 require a flexible ASIC-strength synthesis solution. The advanced optimization engine in Synopsys Design Compiler FPGA efficiently utilizes the special resources available in Virtex-4 devices to provide the highest performance design possible.

DC FPGA gives you the freedom to modify synthesis scripts to address design bottlenecks, implement different

FSM encoding styles, or to explore other design optimizations to reach your design goals. Now you have access to the power and flexibility of Design Compiler to implement your complex FPGA designs.

DC FPGA is an integral part of the complete ASIC-strength prototyping solution from Synopsys. Other tools supported in the Xilinx flow are Formality™ for formal verification, DesignWare® Library IP, Leda® for RTL design and code checking, PrimeTime® for static timing analysis, VCS® for simulation, Module Compiler™ for datapath synthesis, and HSPICE™ for analysis of multi-gigabit serial I/Os.

DC FPGA has a rapidly growing base of more than 100 customers. For more information about Design Compiler FPGA, visit www.synopsys.com/products/dcfpga/dcfpga.html.

Design	Test Description	Implementation with DSP48 Max Delay (ns)	Implementation without DSP48 Max Delay (ns)
Test1*	A_reg[17:0] (FD) <= A B_reg[17:0] (FD) <= B Q[35:0] (FD) <= A_reg * B_reg	3.028	3.062
Test2	A_reg1[16:0] (FD) <= A_reg (FD) <= A B_reg1[16:0] (FD) <= B_reg (FD) <= B mult[34:0] (FD) <= A_reg1 * B_reg1 Q[47:0] (FD) <= Q + mult	1.720	5.444
Test3	Q[47:0] (FD) <= Q + A[16:0] * B[16:0]	3.954	7.975
Test4	A_reg[16:0] (FD) <= A B_reg[16:0] (FD) <= B C_reg[47:0] (FD) <= C Q[47:0] (FD) <= sel ? C_reg + (A_reg * B_reg) : C_reg - (A_reg * B_reg)	1.633	8.081
Test5	A[16:0], B[16:0], C[47:0] Q[47:0] = sel ? C + (A * B) : C - (A * B)	5.680	8.177
Test6	A[16:0], B[16:0], C[16:0], D[16:0] E[16:0], F[16:0], G[16:0], H[16:0] mult1[33:0] (FD) <= A * B + C * D mult2[33:0] (FD) <= E * F + G * H Q[47:0] = mult1 + mult2	6.151	7.631

* Input and output signals are signed

Table 2 - Design examples showing performance improvement of advanced DC FPGA DSP inference

Selecting Connectors for Multi-Gigabit Transceiver Designs

With data transfer rates at 10 Gbps, connector choice is crucial.

by Marc Defossez
Sr. Staff Applications Engineer
Xilinx, Inc.
marc.defossez@xilinx.com

In modern high-speed digital designs, connectors require careful attention; you can't just use any one that's available. When designing with Xilinx® Virtex-4™ multi-gigabit transceiver (MGT) devices, with data transfer rates increasing to 10 Gbps, connectors are part of the total solution.

It is often said that the silicon, in our case the FPGA, does all the work in a system. Passive components such as connectors get the blame for increasing design cost, complexity, and size, and therefore are often neglected.

Today's digital designs enter the RF world with transfer speeds of 10 Gbps and more per data pair; thus, you can no longer ignore the overall impact connector choice has on a design.

Connector manufacturers must keep track of high-speed digital design needs while meeting the demand for multiple high-speed low-loss connections in a small connector shape. Connector design, therefore, becomes increasingly difficult.

The two worlds need to be combined; therefore, we advise following these steps when selecting a connector:

- Choose your connector type – backplane, board-to-board, board-to-cable, or mezzanine
- Find manufacturers carrying connectors with the right physical parameters
- Carefully examine the manufacturer's electrical specifications, test reports, and other published references

Board-to-Backplane or Board-to-Board Connectors

Designing a system in which multiple MGT signals (3.125 Gbps to 10 Gbps) cross directly from board to board or run over a backplane need special connectors. The Teradyne™ GBx connector is a high-density, optimized differential connector family delivering data rates greater than 5 Gbps (tested up to 12 Gbps) (Figure 1).

Tyco™-AMP offers in this same range the Z-Pack HM-Zd differential connector system, designed for serial switching applications from 3.125 Gbps to 6.4 Gbps (demonstrated at 12 Gbps) (Figure 2).

Both connector families are made specifically for high-data-transfer-rate designs such as enterprise switching equipment, telecommunications equipment, and mass data storage. They are robust, have a modular setup, and offer routability and optimal system performance.

Teradyne's GbX advanced performance interconnects provide high-density optimized differential connectors. They are available in three-, four-, and five-pair versions and permit vertical and horizontal routing, making them the ideal solution for star or mesh backplane designs.

Tyco-AMP's high-speed, differential, board-to-backplane electrical connectors are an extension of the already established IEC 61076-4-101 hard metric connector family. However, HM-Zd also provides a high-speed differential solution. Z-Pack HM-Zd connectors are available in two-, three-, and four-paired versions.

In board-to-board designs where size matters, Samtec's™ QSE and QTE connector families are for data transfer rates up to 6 Gbps (Figure 3).

For board to board, with a point-to-point setup, Samtec offers a reliable cable connection based on the QSE/QTE con-

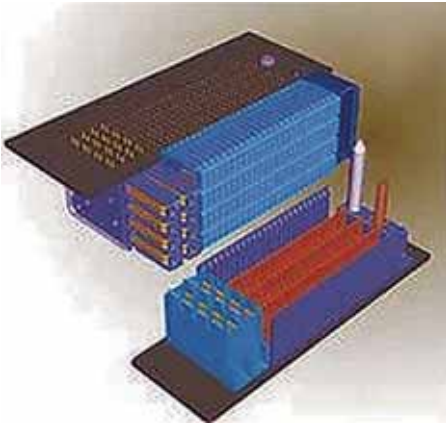


Figure 1 – Teradyne Gbx connector

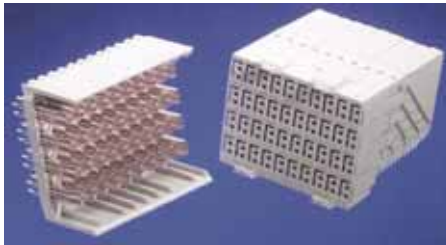


Figure 2 – Tyco-AMP Z-Pack HM-Zd connector

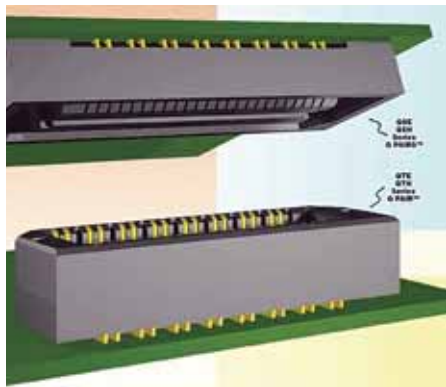


Figure 3 – Samtec QSE and QTE connector



Figure 4 – Samtec ribbon cable

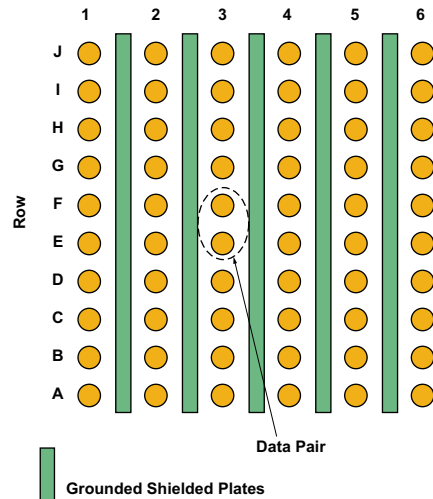


Figure 5 – Stripline construction of NexLev

nector technology. The 50 ohm controlled impedance, 38 AWG mini coax ribbon cable (Figure 4) is available with as many as 240 signal lines, as well as a differential or single-ended flex-strip solution.

You can create custom connector specifications for both the QSE/QTE and ribbon cable on Samtec's website and download cable specifications and test reports on cross-talk, travel delay, and impedance.

Mezzanine Board-to-Board Connectors

Mezzanine card systems are mostly used to relocate high-pin-count devices onto mezzanine or module cards, simplifying board routing without compromising system performance.

Mezzanine cards need a high bandwidth and high amount of parallel connections as well as several serial connections. Teradyne's version is the NexLev connector family, with performance up to 12 Gbps. This connector enables a vast amount of connection possibilities at different connector heights.

The NexLev connector is built in a stripline construction, providing a continuous ground plane for each signal contact (Figure 5). The connectors come as ten-row connectors with 100, 200, or 300 positions at possible stacking heights from 10 mm to 30 mm. You can find technical figures at www.teradyne.com/prods/tcs/products/connectors/mezzanine/nexlev/signintegr.html#differential.

Samtec offers a similar solution with its

YFS/YFT single-ended and differential-pair-array connector arrays called SamArray (Figure 6). These connectors have a performance up to 10 Gbps and comprise a vast amount of single-ended connections. Differential signaling is obtained through pin layout (Figure 7).

Connectors are offered as five-, eight-, or ten-row with as many as 50 contacts per row, for stacking heights from 5 to 25 mm. Technical figures are provided in PDF format at www.samtec.com/signal_integrity/technical_specifications/electrical.asp?series=YFS-DP&stack=25&menu=Signal_Integrity.

Mezzanine connectors have a BGA footprint and can be treated by assembly machines as regular BGA components. Experience with these connectors showed that before soldering, they are best glued to the PCB. If not glued, there is a great chance that the connector will move during soldering.

Connectors for Cable Connections

For design reasons you may not be able to use the connectors described above. In this case you can still turn to older solutions, such as the well-known SMA connector and the small MMCX connector.

SMA is an acronym for "SubMiniature version A," first developed in the 1960s. They are 50 ohm, semi-precision subminiature units that provide excellent electrical performance from DC to 18 GHz with a threaded interface. These high-performance connectors are compact in size and have outstanding mechanical specifications.

Besides the standard straight, 90 degrees, and edge-launch version, an SMT-mount device version is now also available (Figure 8). This SMT version is preferable over the other because of its performance characteristics.

The MMCX series is sometimes also called MicroMate. It is the smallest RF connector and was developed in the 1990s. MMCX is a micro-miniature connector series with a lock-snap mechanism, allowing for 360 degrees rotation and thus enabling great flexibility in PCB layouts. MMCX connectors conform to the European CECC 22000 specification.

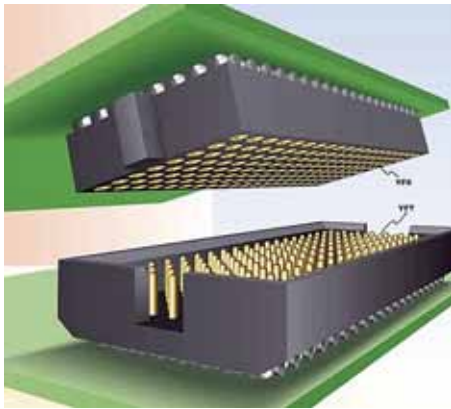


Figure 6 – Samtec YFS/YFT connector

MMCX products range to 6 GHz for a 50 Ω interconnect system. A set of connectors includes surface mount, edge card, and cable connectors. Here the SMT version is preferable (Figure 9).

You can purchase ready-made, custom, and length-matched cable interconnect for this type of connection from different sources and choose between flexible or semi-rigid cabling.

Connector Basics

Suppose you've selected your IC devices and your board has been laid out with all of

You're not quite done yet. In high-performance systems, every element must be optimized for the entire system to meet performance, schedule constraints, size, and cost. It is like a chain – every link must be strong for the whole to meet the demanding performance specs of today's high-speed products.

How can components like connectors affect system performance? Usually the potential problems are lumped into two categories: timing and noise, together referred to as signal integrity (SI).

What is important when selecting connectors?

- EMI, translated to series inductance
- Crosstalk, translated to mutual inductance
- Signal propagation, as parasitic capacitance

Series Inductance

The most fundamental effect a connector adds to a circuit is series inductance. The primary factor for the series inductance is the pin length of the connector. Together with the series inductance of each connector pin, the pin layout of the connector determines the radiated EMI (electromagnetic interference).

Signals traveling through a connector need a current return path (ground). Even if no return path is provided through the connector, large inductive loops can be created (Figure 10). This will result in substantial EMI emission.

Differential signaling solves the problem of current return paths by eliminating it. Differential signaling uses two identical but opposite signals. The return paths are therefore also opposite to each other (Figure 11). This effect will cancel out. The only signal returning from a differential pair is because of an imbalance between the two signals. The subtraction of both signals will not be exactly zero.

Mutual Inductance

Current loops illustrate mutual inductive coupling in Figure 12. Current leaving

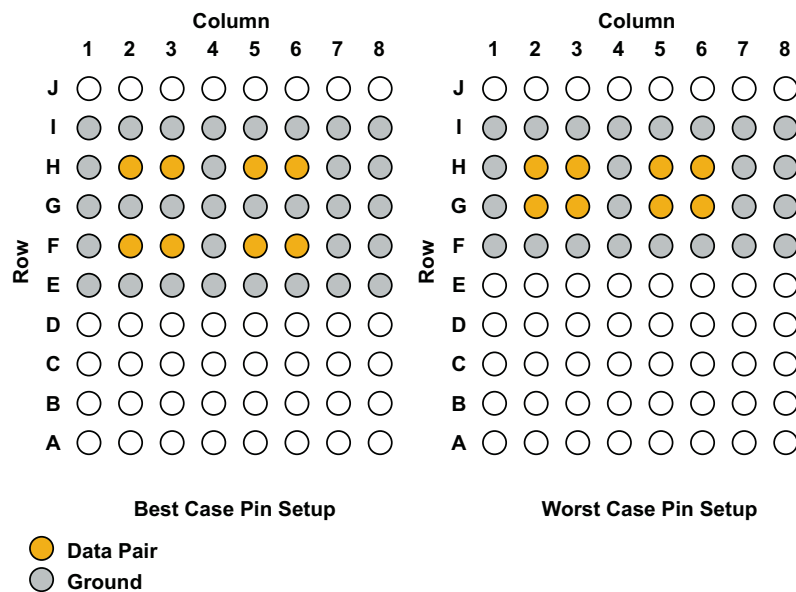


Figure 7 – Best- and worst-case pin layout for YFS/YFT



Figure 8 – SMA edge launch, SMT



Figure 9 – MMCX edge launch, SMT

the right design rules, such as:

- Controlled impedance traces
- Controlled time delay of stubs
- Stubs shorter than about 20% of the fastest signal's rise time
- Time delay of discontinuities shorter than about 15% of the fastest signal's rise time.
- Adjacent traces paced far enough apart to keep crosstalk at an acceptable level
- A stack-up with power and ground planes on adjacent layers of silicon
- A continuous return path under each signal trace

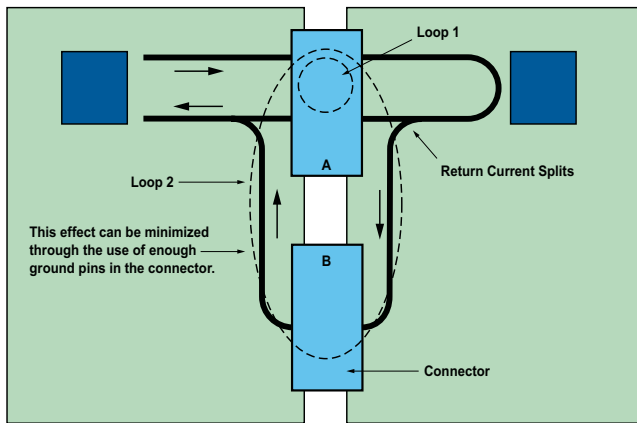


Figure 10 – EMI generated due to improper current return paths

device A returns through signal return path X. Even currents leaving devices B and C have signal return paths through Y and Z.

Because all of these paths overlap, magnetic fields from one path induce electric voltages (noise) in other paths. The induced noise will be larger or smaller with the physical location of a path. In our example, Y will receive more noise than Z because it shares more area.

Do not worry about crosstalk between differential signals. Because of their nature, crosstalk is canceled out.

Parasitic Capacitance

Mutual and shunt (pin-to-pin) capacitance is another effect that comes with a connector – usually you can ignore it. The effect capacitance has is to slow down system edge rate. In multi-drop backplane applications, parasitic capacitance places more burdens on connectors than in point-to-point applications.

Signals transmitted pass each tap on the bus; the cumulative effect of the parasitic capacitance can distort the signals and the series inductance of the source connector.

Connector Selection

To provide excellent high-speed connectors, manufacturers need to control and manage the above parameters as well as a lot more. Engineers now have access to an extensive amount of data measured and calculated by connector manufacturers.

On most manufacturers' websites, electrical, mechanical, and SI information is

available, together with PCB drawing and simulation aids:

- Mechanical
 - Dimension drawing in PDF format
 - 3D models in IGES, STEP, or Parasolid ACIS format
 - Mechanical qualification and stress test reports
 - PCB layout tool library components
- Electrical
 - Electrical test reports
 - Application notes
 - SI parameters and results
 - Datasheets
- Simulation
 - IBIS and SPICE models

An extra service offered by Samtec is the "Final Inch" website, for designing a connector break-out region on a PCB.

The manufacturers mentioned in this article are not the only high-speed connector manufacturers on the market. There are other companies such as ERNI™, Hirose, Molex™, Amphenol™, and Radiall™ manufacturing (under license) similar con-

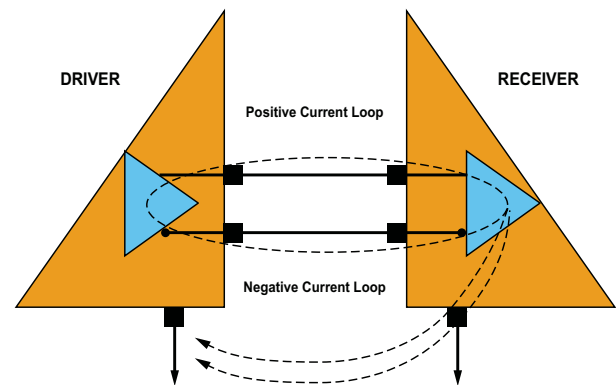


Figure 11 – Differential eliminated returned signal currents

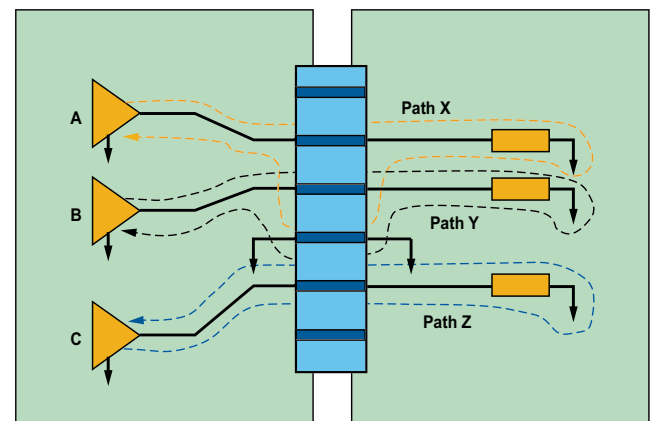


Figure 12 – Mutual inductive coupling through a connector

nectors. Many other companies have their own range of high-speed connectors.

Conclusion

Today's high-speed digital design engineers can benefit from the RF knowledge of connector suppliers, using the information available in datasheets, application notes, and on the Internet.

You can use this article as a starting point for better PCB and connector design.

For more information, see the books "High-Speed Digital System Design" by Stephen H. Hall, Garrett W. Hall, and James A. McCall; "High-Speed Digital Design" by Howard Johnson; or visit www.johnson-comp.com, www.samtec.com, www.samtec.com/sudden_service/current_literature/q-pairs/index.html, www.samtec.com/sudden_service/current_literature/SamArray/index.html, www.teradyne.com/prods/tcs, and hmdz.tycoelectronics.com.

Xilinx/Micron Partner to Provide High-Speed Memory Interfaces

Micron's RLDram II and DDR/DDR2 memory combines performance-critical features to provide both flexibility and simplicity for Virtex-4-supported applications.

by Mike Black
Strategic Marketing Manager
Micron Technology, Inc.
mblack@micron.com

With network line rates steadily increasing, memory density and performance are becoming extremely important in enabling network system optimization. Micron Technology's RLDram™ and DDR2 memories, combined with Xilinx® Virtex-4™ FPGAs, provide a platform designed for performance.

This combination provides the critical features networking and storage applications need: high density and high bandwidth. The ML461 Advanced Memory Development System (Figure 1) demonstrates high-speed memory interfaces with Virtex-4 devices and helps reduce time to market for your design.

Micron Memory

With a DRAM portfolio that's among the most comprehensive, flexible, and reliable in the industry, Micron has the ideal solution to enable the latest memory platforms. Innovative new RLDram and DDR2 architectures are advancing system designs farther than ever, and Micron is at the forefront, enabling customers to take advantage of the new features and functionality of Virtex-4 devices.

RLDRAM II Memory

An advanced DRAM, RLDram II memory uses an eight-bank architecture optimized for high-speed operation and a double-data-rate I/O for increased bandwidth. The eight-bank architecture enables

RLDRAM II devices to achieve peak bandwidth by decreasing the probability of random access conflicts.

In addition, incorporating eight banks results in a reduced bank size compared to typical DRAM devices, which use four. The smaller bank size enables shorter address and data lines, effectively reducing the parasitics and access time.

Although bank management remains important with RLDram II architecture, even at its worst case (burst of two at 400 MHz operation), one bank is always available for use. Increasing the burst length of the device increases the number of banks available.

I/O Options

RLDRAM II architecture offers separate I/O (SIO) and common I/O (CIO) options. SIO devices have separate read and write ports to eliminate bus turnaround cycles and contention. Optimized for near-term read and write balance, RLDram II SIO devices are able to achieve full bus utilization.

In the alternative, CIO devices have a shared read/write port that requires one additional cycle to turn the bus around. RLDram II CIO architecture is optimized for data streaming, where the near-term bus operation is either 100 percent read or 100 percent write, independent of the long-term balance. You can choose an I/O version that provides an optimal compromise between performance and utilization.

The RLDram II I/O interface provides other features and options, including support for both 1.5V and 1.8V I/O lev-

els, as well as programmable output impedance that enables compatibility with both HSTL and SSTL I/O schemes. Micron's RLDram II devices are also equipped with on-die termination (ODT) to enable more stable operation at high speeds in multipoint systems. These features provide simplicity and flexibility for high-speed designs by bringing both end termination and source termination resistors into the memory device. You can take advantage of these features as needed to reach the RLDram II operating speed of 400 MHz DDR (800 MHz data transfer).

At high-frequency operation, however, it is important that you analyze the signal driver, receiver, printed circuit board network, and terminations to obtain good signal integrity and the best possible voltage and timing margins. Without proper terminations, the system may suffer from excessive reflections and ringing, leading to reduced voltage and timing margins. This, in turn, can lead to marginal designs and cause random soft errors that are very difficult to debug. Micron's RLDram II devices provide simple, effective, and flexible termination options for high-speed memory designs.

On-Die Source Termination Resistor

The RLDram II DQ pins also have on-die source termination. The DQ output driver impedance can be set in the range of 25 to 60 ohms. The driver impedance is selected by means of a single external resistor to ground that establishes the driver impedance for all of the device DQ drivers.

As was the case with the on-die end termination resistor, using the RLDram II



on-die source termination resistor eliminates the need to place termination resistors on the board – saving design time, board space, material costs, and assembly costs, while increasing product reliability. It also eliminates the cost and complexity of end termination for the controller at that end of the bus. With flexible source termination, you can build a single printed circuit board with various configurations that differ only by load options, and adjust the Micron RDRAM II memory driver impedance with a single resistor change.

DDR/DDR2 SDRAM

DRAM architecture changes enable twice the bandwidth without increasing the demand on the DRAM core, and keep the power low. These evolutionary changes enable DDR2 to operate between 400 MHz and 533 MHz, with the potential of extending to 667 MHz and 800 MHz. A summary of the functionality changes is shown in Table 1.

Modifications to the DRAM architecture include shortened row lengths for reduced activation power, burst lengths of four and eight for improved data bandwidth capability, and the addition of eight banks in 1 Gb densities and above.

New signaling features include on-die termination (ODT) and on-chip driver (OCD). ODT provides improved signal quality, with better system termination on the data signals. OCD calibration provides the option of tightening the variance of the pull-up and pull-down output driver at 18 ohms nominal.

Modifications were also made to the mode register and extended mode register, including column address strobe CAS latency, additive latency, and programmable data strobes.

Conclusion

The built-in silicon features of Virtex-4 devices – including ChipSync™ I/O technology, SmartRAM, and Xesium differential clocking – have helped simplify interfacing FPGAs to very-high-speed memory devices. A 64-tap 80 ps absolute delay element as well as input and output DDR registers are available in each I/O element, providing for the first time a run-time center alignment of data and clock that guarantees reliable data capture at high speeds.



Figure 1 – ML461 Advanced Memory Development System

Xilinx engineered the ML461 Advanced Memory Development System to demonstrate high-speed memory interfaces with Virtex-4 FPGAs. These include interfaces with Micron's PC3200 and PC2-5300 DIMM modules, DDR400 and DDR2533 components, and RDRAM II devices.

In addition to these interfaces, the ML461 also demonstrates high speed QDR-II and FCRAM-II interfaces to

Virtex-4 devices. The ML461 system, which also includes the whole suite of reference designs to the various memory devices and the memory interface generator, will help you implement flexible, high-bandwidth memory solutions with Virtex-4 devices.

Please refer to the RDRAM information pages at www.micron.com/products/dram/rldram/ for more information and technical details. 🌈

FEATURE/OPTION	DDR	DDR2
Data Transfer Rate	266, 333, 400 MHz	400, 533, 667, 800 MHz
Package	TSOP and FBGA	FBGA only
Operating Voltage	2.5V	1.8V
I/O Voltage	2.5V	1.8V
I/O Type	SSTL_2	SSTL_18
Densities	64 Mb-1 Gb	256 Mb-4 Gb
Internal Banks	4	4 and 8
Prefetch (MIN Write Burst)	2	4
CAS Latency (CL)	2, 2.5, 3 Clocks	3, 4, 5 Clocks
Additive Latency (AL)	No	0, 1, 2, 3, 4 Clocks
READ Latency	CL	AL + CL
WRITE Latency	Fixed	READ Latency - 1 Clock
I/O Width	x4/ x8/ x16	x4/ x8/ x16
Output Calibration	None	OCD
Data Strobes	Bidirectional Strobe (Single-Ended)	Bidirectional Strobe (Single-Ended or Differential) with RDQS
On-Die Termination	None	Selectable
Burst Lengths	2, 4, 8	4, 8

Table 1 – DDR/DDR2 feature overview

Harvesting the Flexibility of Virtex-4 RocketIO Transceivers

New features include support for all major serial I/O standards and multiple encoding schemes.

by Matt DiPaolo
APD Product Application Engineer
Xilinx, Inc.
matt.dipaolo@xilinx.com

Ryan Carlson
Director of Marketing, High Speed Serial I/O
Xilinx, Inc.
ryan.carlson@xilinx.com

Xilinx® introduced FPGAs with integrated multi-gigabit serial transceivers (MGTs) more than three years ago. Since then, Virtex-II Pro™ devices have enabled hundreds of applications to move from parallel interfaces to high-speed serial interfaces, as designers took advantage of the integrated RocketIO™ transceivers.

With Virtex-II Pro devices, Xilinx led the industry with a transceiver capable of 622 Mbps-3.125 Gbps operation. Xilinx continues this trend with its new Virtex-4™ family, in which RocketIO transceivers can operate from 622 Mbps to over 10 Gbps (Figure 1). This broad speed range – coupled with a host of user-friendly, programmable options – creates an extremely flexible multi-gigabit transceiver.

Multiple Interface Standards

One trend occurring in multiple end-market segments is the widespread adoption of high-speed differential signaling schemes to address increased bandwidth demands. As designs move to faster interface speeds, a serial implementation saves power, board space, design complexity, and ultimately cost.

Virtex-4 RocketIO transceivers were designed to enable high-speed data transmission for many different protocols. Table 1 shows all of the serial standards supported in Virtex-4 FPGAs.

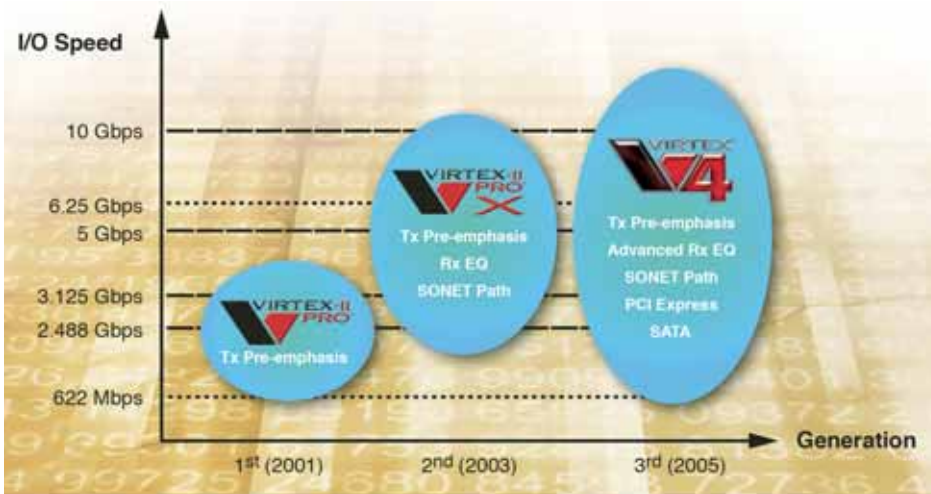


Figure 1 – Evolution of the RocketIO transceiver

Flexibility and Programmability

Xilinx brings its approach to FPGAs – making them user-programmable, with maximum flexibility – to its multi-gigabit transceivers. This approach has impacted both of the major functional components of the RocketIO transceiver: the physical media attachment (PMA) block and the physical coding sublayer (PCS) block.

PMA Block

The Virtex-4 RocketIO PMA block supports all major serial I/O standards and is compliant to their physical layer requirements. For example, the RocketIO transceiver meets the OC-48 SONET/SDH specification (2.488 Gbps) for both transmit jitter generation and receive jitter tolerance.

This same transceiver can also meet the requirements of the Fibre Channel physical layer specification, and it can do so at 1.0625 Gbps, 2.125 Gbps, 4.25 Gbps, and 8.5 Gbps.

Other PMA features of the Virtex-4 RocketIO transceiver include:

- Programmable transmit pre-emphasis (3-tap)
- Programmable active receive equalization
- Programmable decision-feedback equalization (DFE)
- Integrated receiver AC-coupling capacitors (user-bypassable)

- PCI Express-compliant electrical idle support
- PCI Express-compliant beaconing support
- PCI Express-compliant spread spectrum clocking support
- Multiple loopback modes, including a PMA Rx to Tx path

PCS Block

The Virtex-4 RocketIO PCS block supports multiple encoding schemes; both 8B10B and 64B66B encoders/decoders are

built into the transceiver. You can select a 10-bit based data path (for Ethernet and data communications protocols) or a 16-bit based data path (for SONET/SDH-based protocols).

User-programmable clock correction sequences (CCS) allow synchronization differences between remote transceivers to be tolerated and corrected. Channel bonding sequences (CBS) enable you to connect multiple RocketIO transceivers together to create a logical channel with even more bandwidth. All of these features are compliant to industry standards (making designs easier to complete), while still supporting proprietary designs.

For applications requiring lower latency, a new feature of the Virtex-4 RocketIO transceiver is a reduced latency mode that allows you to bypass the receive and transmit FIFOs (as well as other function blocks), offering a 50% reduction in latency from previous generations of Xilinx transceivers.

Other PCS features of the Virtex-4 RocketIO transceiver include:

- Multiple loopback modes, including a PMA Rx to Tx path
- Comma detection, including A1A1A2A2 for SONET applications

Mode	Channels (Lanes)	I/O Bit Rate (Gbps)
SONET OC-12	1	0.622
Fibre Channel (1, 2, 4, 8)	1	1.0625/2.125/4.25/8.5
Gb Ethernet	1	1.25
SONET OC-48	1	2.488
Infiniband	1/4/12	2.5
PCI Express	1/2/4/8/16	2.5
Serial Rapid IO	1	1.25/2.5/3.125
Serial ATA	1	1.5/3
XAUI (10 Gb Ethernet)	4	3.125
XAUI (10 Gb Fibre Channel)	4	3.1875
SONET OC-192	1	9.95328
10 Gb Ethernet	1	10.3125

Table 1 – Example supported standards of the Virtex-4 RocketIO transceiver

- Clock correction/channel bonding receive elastic buffer
- Autonomous CRC-32 blocks (one for transmitted data and one for received data)
- Dynamic configuration bus to access every PCS attribute dynamically, including CCS and CBS
- 64B66B block sync, gearbox, encoder/decoder, and scrambler/descrambler
- 8B10B encoder/decoder
- Built-in clock dividers to reduce the need of DCMs for clocking use models

Figures 2 and 3 show block diagrams of the Virtex-4 PCS (both receiver and transmitter).

Conclusion

The Virtex-4 RocketIO transceiver is the complete solution for today's high-speed serial designs, with a broad speed range (622 Mbps to over 10 Gbps) and programmable PCS functions (optional encoding schemes, channel bonding, and clock correction).

For more information about the Virtex-4 FPGA family, visit www.xilinx.com/virtex4/. For more details about the functionality and design recommendations with Virtex-4 RocketIO transceivers, see the Virtex-4 RocketIO transceiver user guide at www.xilinx.com/bvdocs/userguides/ug076.pdf.

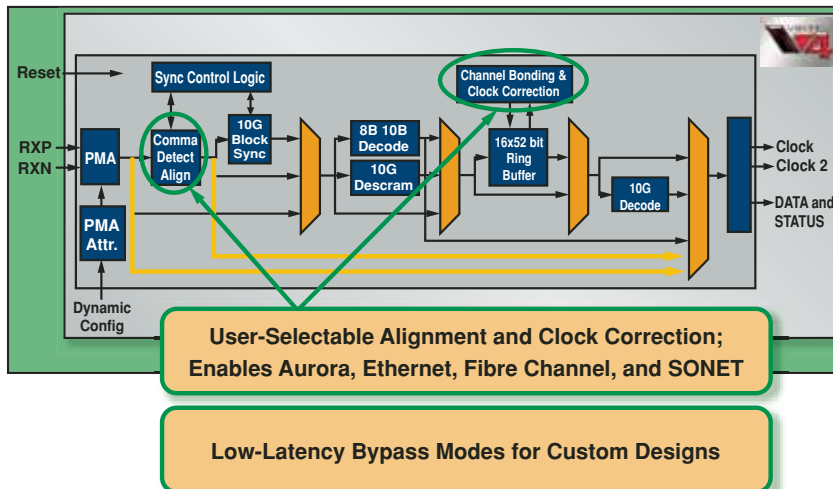


Figure 2 – Virtex-4 RocketIO PCS (receiver)

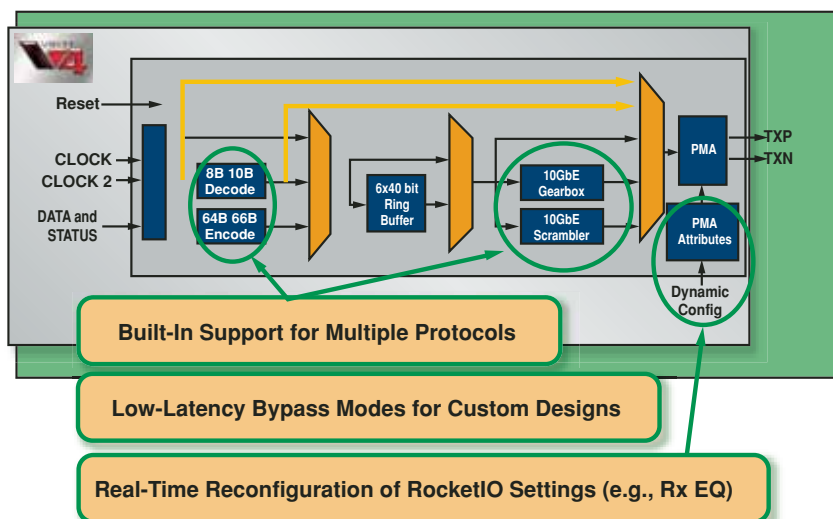


Figure 3 – Virtex-4 RocketIO PCS (transmitter)

Xilinx Events and Tradeshow

Xilinx participates in numerous trade shows and events throughout the year. This is a perfect opportunity to meet our silicon and software experts, ask questions, see demonstrations of new products and technologies, and hear other customers' success stories with Xilinx products. For more information and the most up-to-date schedule, visit: www.xilinx.com/events/.

Worldwide Events Schedule

North America

- Jan. 31 - Feb. 3 DesignCon West
Santa Clara, CA
- February 15-17 TI Developers Conference
Houston, TX
- March 1-3 Intel Developer Forum
San Francisco, CA
- March 8-10 Embedded Systems Conference
San Francisco, CA

Europe

- Jan. 31 - Feb. 2 Elektronik Systeme im Automobil
Munich, Germany
- February 1-3 EP05 Electronic Exhibition
Stockholm, Sweden
- February 14-17 3GSM World Congress
Cannes, France
- February 22-24 Embedded World
Nuremberg, Germany
- March 16-17 Workshop SoC Défense
Brussels, Belgium
- March 16-17 Hi-Tech Technologies
Tel Aviv, Israel
- March 17-18 AMAA Conference and Exhibition
Berlin, Germany

Japan

- January 29-30 EDSF
Yokohama, Japan
- February 15 Processor Seminar
Osaka, Japan
- February 21 Processor Seminar
Tokyo, Japan



Optimize Memory Subsystem Performance with Network FCRAM

Toshiba's Network FCRAM often provides the best cost/performance by combining DRAM densities with random cycle performances that approach SRAM speeds.

by Scott Beekman
Business Development Manager
Toshiba America Electronic Components, Inc.
scott.beekman@taec.toshiba.com

Among the many cost/performance trade-offs system designers face, one of the critical decisions in network systems, communications equipment, and high-performance consumer electronics is the type of memory to use to ensure that performance can keep pace with the processor.

Traditionally, network system designers had to choose between dynamic random access memory (DRAM), available at a lower cost-per-bit because of the high volumes used in personal computers, or higher performance static random access memory (SRAM), available only in low densities and at a much higher cost. A combination of the two is typically used with DRAM for buffer memory and SRAM for look-up table (LUT) memory.

More recently, high-performance, low-latency DRAM solutions developed specifically for high-bandwidth applications, including Toshiba's™ Network FCRAM™ (fast cycle random access memory), provide another alternative. Which type of memory is right for your particular system? What additional requirements for memory controllers are associated with each choice?

Generally, you can choose the option that provides the highest performance within the system's specified cost constraints, and in the time available to bring the system to market. In many cases, Network FCRAM provides the best cost/performance for networking and communications customers by combining DRAM densities with random cycle performances that approach SRAM speeds. This allows equipment manufacturers to develop higher performance, lower cost, and lower power communications systems than they could with double-data-rate synchronous dynamic RAM (DDR SDRAM) and high-speed static RAM (HSSRAM).

In this article, we provide an overview of Network FCRAM and the advantages it offers in comparison to standard DDR SDRAM or high-speed SRAM, and discuss the alternatives available for memory controllers supporting Network FCRAM.

Network FCRAM

Toshiba Network FCRAM is a high-performance, low-cost replacement to DDR SDRAM and high-speed SRAM targeted primarily for buffer memory and LUT memory in networking/telecom applications. Network FCRAM incorporates enhanced DRAM technology optimized for the high-bandwidth, low-latency requirements of network and communication systems. Narrowing the active memory area achieves low power consumption and random cycle time performances almost triple that of standard DRAM.

Network FCRAM devices offer the following advantages:

- Fast random cycle time (t_{RC}) of 20 ns to 25 ns
- Fast data transfer rate of 666 Mbps+ (For purposes of measuring data transfer rate in this context, megabit per second and/or Mbps = 1,000,000 bits per second.)

- Large density up to 512 Mb (When used in relation to memory density, megabit and/or Mb means 1,024 x 1,024 = 1,048,576 bits. Usable capacity may be less. For details, please refer to specifications.)
- Simplified command input
- Low power consumption
- Multiple sources

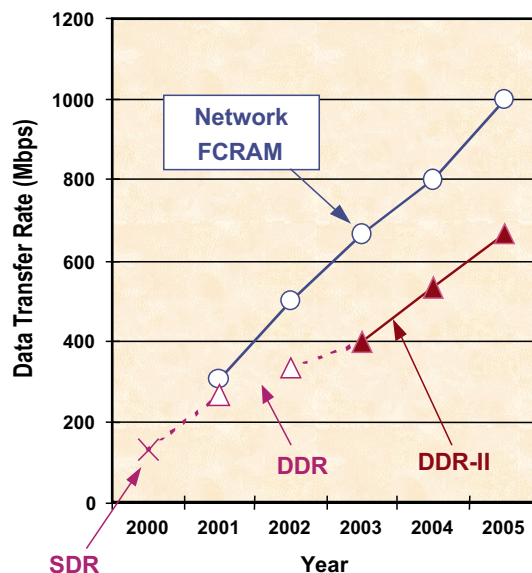


Figure 1 – Faster data transfer rates with Network FCRAM

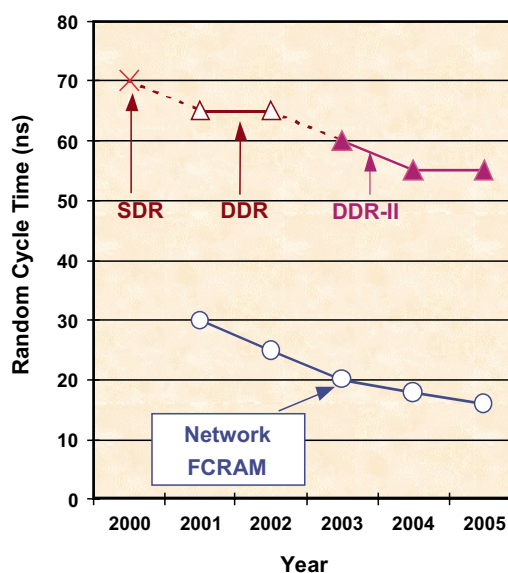


Figure 2 – Network FCRAM typically provides 20 to 25 percent higher system performance than DDR SDRAM offers, in part because of its faster random cycle time (approximately three times faster).

Network FCRAM technology excels in applications where you need DRAM densities and random cycle performance approaching SRAM-like speeds. Its high bandwidth and low latency makes Network FCRAM suitable for network applications, cache applications, and high-performance consumer applications. Typical network equipment applications include packet buffer memory, table look-up memory, and external cache memory in servers. Network FCRAM is also being used in digital consumer and supercomputer applications.

Performance Comparison

Network FCRAM and the specification-compatible, dual-source Samsung™ Network DRAM™ feature one of the shortest cycle times and latency among existing DRAM. As a result, Network FCRAM can improve system performance approximately 20 to 25 percent in comparison to DDR SDRAM. This is achieved as a result of higher data transfer rates, as shown in Figure 1, and an approximately threefold faster random cycle time (t_{RC}), as shown in Figure 2.

As an alternative to HSSRAM, Network FCRAM costs approximately 1/16th as much per bit, and offers much higher densities (up to 512 Mb) compared to maximum densities of 36 Mb or 72 Mb for HSSRAM. Network FCRAM offers not only performance improvement alternatives but also lower-cost solutions, as shown in Figure 3.

Customers today are taking advantage of these features to boost performance and bring down their system's cost by replacing DDR SDRAM with Network FCRAM, thus reducing chip count and board space because of Network FCRAM's higher performance, and/or by replacing HSSRAM.



Selecting the Right FCRAM

Network FCRAM is available with a selection of interfaces, speeds, and organizations to meet various requirements:

- 256 Mb (x8/ x 16) Network FCRAM1 (up to 400 Mbps with $t_{RC} = 25$ ns)
- 288 Mb (x18) Network FCRAM2 (up to 666 Mbps with $t_{RC} = 20$ ns)
- 288 Mb (x36) Network FCRAM2 (up to 666 Mbps with $t_{RC} = 20$ ns)
- 512 Mb (x8/ x 16) Network FCRAM1 (up to 533 Mbps with $t_{RC} = 22.5$ ns)

Network FCRAM1 supports non-ECC bit densities (such as 256 Mb and 512 Mb as a single component), while Network FCRAM2 supports ECC bit densities (such as 288 Mb with roadmaps to higher densities).

Memory Controllers

Once you have selected Network FCRAM as the memory of choice for a design, the next step is to determine the best source of a memory controller for your system. For large-volume applications, some customers develop custom ASICs that include the memory controller; in addition, many network processors (NPUs) now support Network FCRAM. However, for many smaller volume applications, FPGAs offer lower cost and faster time to market.

Xilinx® Virtex-II™, Virtex-II Pro™, and Virtex-4™ FPGAs interface to Network FCRAM.

When evaluating memory alternatives for network systems, consider the performance advantages of Network FCRAM and the time-to-market advantages of an FPGA-based memory controller.

Development Tools

Toshiba offers several design guides to help customers and systems architects identify the key advantages of incorporating Network FCRAM technology into their high-performance applications. Network

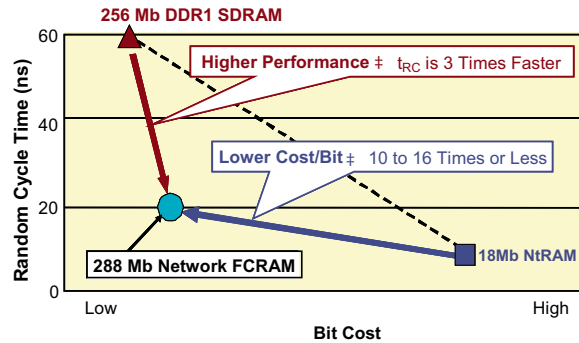


Figure 3- Network FCRAM can also be a lower cost alternative to HSSRAM, as it costs approximately 1/10th to 1/16th as much per bit.

FCRAM devices are also supported by advanced simulation models to facilitate and accelerate design-in activity. Models supported include Verilog™, HSPICE™ and IBIS models, and SOMA models jointly developed by Toshiba and Denali™ Software Inc. For more information, visit www.fcram.toshiba.com.

Conclusion

As a result of Network FCRAM's cost-performance advantages, today it is designed into more than 100 network solutions at more than 70 companies. Toshiba first introduced Network FCRAM working samples in 1999 and has continued to expand its product offering and build momentum in the network/telecom market.

Today, Network FCRAM is in production with data transfer rates as high as 666 Mbps and random cycle time performance as low as 20 ns. Toshiba now supports three densities in mass production, with higher density, higher bandwidth, and faster devices planned for 2005.

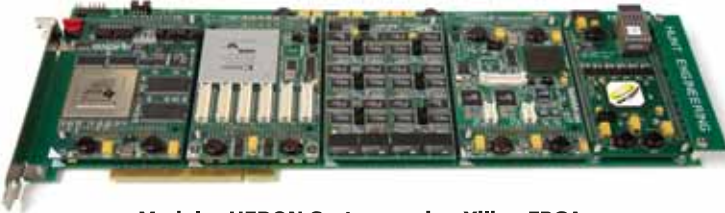
The official Network FCRAM/DRAM website can be found at www.networkfcram.com.

FCRAM (Fast Cycle RAM) is a trademark or a registered trademark of Fujitsu Limited, Japan. Memory Modeler AV is a trademark of Denali Software Inc. Network DRAM is a trademark or a registered trademark of Samsung Electronics Co., Ltd. Korea.



TRAQUAIR
www.traquair.com

The High Performance DSP & FPGA Solution



Modular HERON Systems using Xilinx FPGAs

HERON Systems offer the ultimate in high performance signal processing hardware capabilities, for use in PCI, CompactPCI, and embedded real-time environments.

A modular and extensible hardware architecture allows developers to realize their ideas and address complex system requirements using high density Xilinx FPGA technology.

Xilinx Virtex-II Pro and Virtex-II FPGAs, powerful Texas Instruments TMS320C6000 DSP Processors, ultra-fast A/D, D/A, and user-configurable Digital I/O interfaces can be combined in single and multi-board systems to address application requirements bounded only by the imagination.

Virtex-II™ Pro and Virtex-II™ FPGAs
Single and Multiple FPGA configurations

Optional DSP Processors
Texas Instruments TMS320C6000 DSPs

Multi-Channel Analog & Digital I/O
12-bit A/D up to 210MSPS per channel
16-bit D/A up to 160MSPS per channel

Extensible Configurations
PCI, CompactPCI, Embedded

Learn more at:
www.traquair.com/ads/xcell/heron.html

Traquair Data Systems, Inc. Tel: 607-266-6000 Email: sales@traquair.com Web: www.traquair.com

Virtex-II and Virtex-II Pro are trademarks of Xilinx, Inc.

Using Spartan-3 FPGAs to Implement High-Performance DSP

Spartan-3 FPGAs provide breakthrough cost points for embedded DSP.

by Suhel Dhanani
Sr. Marketing Manager, Spartan Solutions
Xilinx, Inc.
suhel.dhanani@xilinx.com

All low-cost FPGAs provide basic logic capability at attractive prices and serve a broad range of general-purpose design requirements. When you consider embedding DSP functions in an FPGA fabric, however, you may believe that you must choose high-end FPGAs to get platform features such as embedded multipliers and distributed memory.

With Spartan-3™ FPGAs, the landscape for embedded DSP has changed. Spartan-3 devices may be low cost, but they also have the platform features required for DSP designs. These platform features allow area-efficient implementation of signal processing functions – allowing you to realize significantly lower price points.

Spartan-3 devices are ideal as coprocessors or pre-/post-processors, offloading highly computational functions from a programmable DSP to enhance system performance.





Figure 1 – Spartan-3 architecture optimized for lower DSP costs

Optimized for DSP

The Spartan-3 family from Xilinx uses 90 nm process technology in conjunction with 300 mm wafers to dramatically lower the cost of FPGAs. At the same time, the devices incorporate key DSP resources such as embedded 18 x 18-bit multipliers and large blocks (18 kb) of memory, distributed RAM, and shift-register logic. This advanced feature set means that you can use Spartan-3 FPGAs to implement DSP algorithms at a significantly lower cost than competing FPGAs. The specific features that help in efficiently implementing DSP are shown in Figure 1.

In addition to increasing the basic performance of systems, these embedded features enhance device utilization. For instance, the embedded Spartan-3 multiplier would take 300-400 logic elements (LEs) if implemented in the logic fabric. And because the embedded multiplier is adjacent to logic fabric, augmenting the functionality (such as creating accumulators or concatenating the multipliers to create complex arithmetic functions) is fairly straightforward.

Many DSP functions are best implemented in pipelines with time multiplexing for efficiency. This allows you to create faster systems with higher bandwidth, but it comes at the expense of requiring more interim storage elements. For example, a time-multiplexed filter would store the results of individual multiply-accumulate cells in shift registers. Such designs can run

out of registers or memory before they run out of logic resources. The Spartan-3 FPGA family is unique in providing a mode where a single look-up table (LUT)

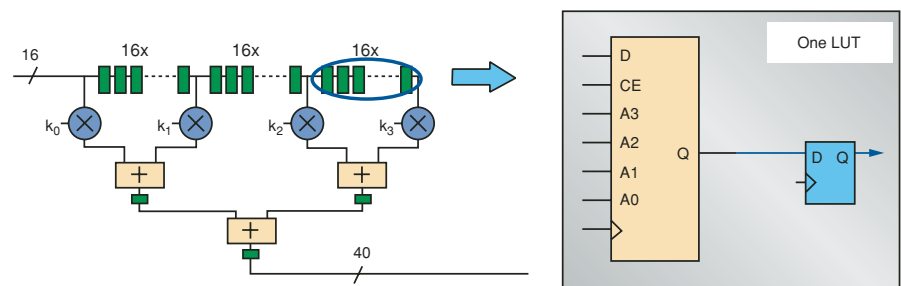


Figure 2 – You can implement 16 registers in one LUT.

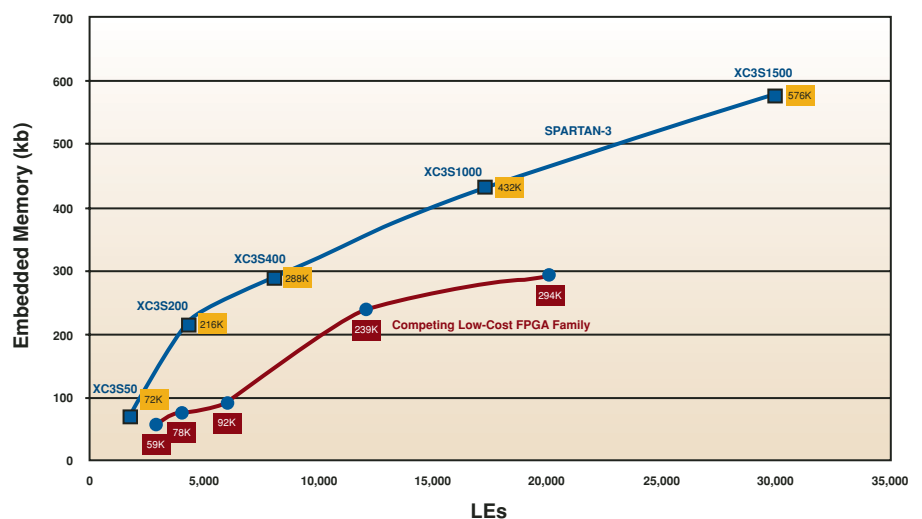


Figure 3 – Spartan-3 fabric provides significantly more memory resources than other competing low-cost FPGAs.

Excerpt from the One-MAC Implementation Report File

Total Number 4 input LUTs:	151 out of	7,168	2%
Number used as logic:	108		
Number used as a route-thru:	13		
Number used as Shift registers:	30		
Number of bonded IOBs:	59 out of	221	26%
IOB Flip Flops:	16		
Number of Block RAMs:	2 out of	16	12%
Number of MULT18X18s:	1 out of	16	6%
Number of GCLKs:	1 out of	8	12%

Excerpt from the Four-MAC Implementation Report File

Total Number 4 input LUTs:	298 out of	7,168	4%
Number used as logic:	162		
Number used as a route-thru:	6		
Number used as Shift registers:	130		
Number of bonded IOBs:	51 out of	173	29%
IOB Flip Flops:	50		
Number of Block RAMs:	4 out of	16	25%
Number of MULT18X18s:	4 out of	16	25%
Number of GCLKs:	1 out of	8	12%

Figure 4 – Using the embedded multipliers and block RAM features of the Spartan-3 fabric for higher performance DSP functions

Common DSP Functions

Let's see how these features impact device utilization by looking at two implementation examples of a finite impulse response (FIR) filter. One is a MAC-based implementation, while the other is a multi-channel distributed arithmetic (DA) implementation.

FIR filters are commonly used in base stations, digital video, wireless LANs, xDSL, and cable modems. Our benchmark is the implementation of a 64-tap, MAC FIR filter with 16-bit data and coefficients running at 130 MHz in a Spartan-3 XC3S400 FPGA. The first implementation uses a single MAC; the second implementation uses four MACs. Figure 4 shows the device utilization section of the report file for both implementations.

Going from a one-MAC to a four-MAC implementation dramatically increases the performance of the FIR filter. The number of LUTs only doubles and remains at just 4% of the total available logic. A four-MAC implementation uses four block RAMs and four multipliers to efficiently implement the FIR filter using minimum device logic resources.

Another interesting implementation is that of a multi-channel FIR function. In this case we can look at how the device utilization changes when we go from a one-channel FIR to an eight-channel FIR filter.

As shown in Figure 5, a single channel distributed arithmetic FIR filter uses 29% of the logic resources and 39% of the registers of a XC3S1000 Spartan-3 device. When implementing an eight-channel version of the same filter, we would normally time multiplex the different channels to conserve logic. But this would use a lot of registers, or a significant amount of on-chip memory to store the intermediate results.

With Spartan-3 FPGAs, the intermediate results are stored in LUTs configured as 16-bit shift registers (SRL-16). This allows the eight-channel version of the same filter to be implemented using only 10% more of the available logic and only 7% more of the available registers – 8x more channels for only 25% more device resources (see Figure 6).

This dramatic savings is directly related to the use of the SRL-16s available in the Spartan-3 device. In the report file, you can see that an additional 1,343 LUTs are used in the SRL-16 mode for the eight-channel implementation.

Implementing this design in an FPGA without SRL16 capability would require an additional 10,744 (1343 x 8) flip-flops used as storage elements, demanding a massive device for the register count and likely squandering the associated combinatorial logic resources.

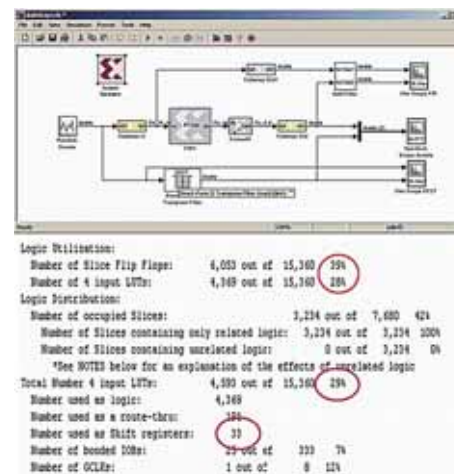


Figure 5 – This single channel DA FIR filter uses 29% of the logic and 39% of the registers in a Spartan-3 XC3S1000 device.

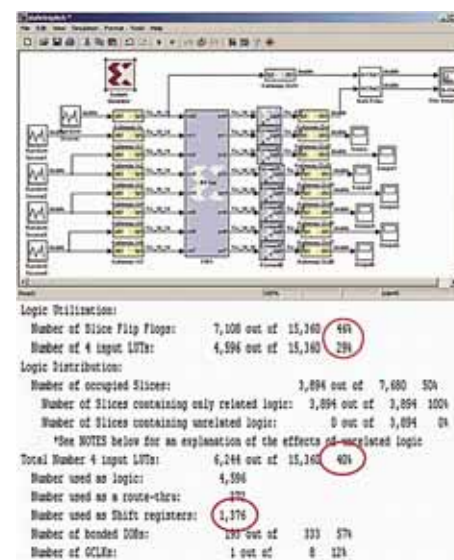


Figure 6 – The eight channel version of the same DA FIR filter only uses 10% more logic and 7% more registers.

Conclusion

The Spartan-3 architecture is optimized to give you very high area efficiency when implementing signal processing functions. By combining these DSP-friendly system features with low unit costs, Spartan-3 FPGAs enable the industry's lowest price points for high-performance DSP functions. This allows a Spartan-3 device to act as a low cost but highly efficient and high-performance co-processor to a programmable DSP processor.

Support Across The Board.™

Get Started Now with Xilinx® Virtex-4™ FPGAs

Xilinx is revolutionizing the fundamentals of FPGA economics with the Virtex-4™ family. To help you get a jumpstart on your next design, Avnet Electronics Marketing has created the Virtex-4 LX Evaluation Kit and a SpeedWay Seminar.™

Virtex-4 FPGAs	Virtex-4LX25 Evaluation Kit
• Multi-Platform FPGA family • Support for (3) application domains • 90 nm process technology • Reduced power consumption • Dramatic reduction in cost per function	• Virtex-4LX25 FPGA • 8 MB Flash and 32 MB DDR SDRAM • Cypress CY7C68013 USB 2.0 controller • National Semiconductor DP83847 10/100 Ethernet PHY • 128x64 OSRAM graphical display

The Virtex-4 SpeedWay Seminar will allow you to:

- Learn about the Virtex-4 product family features
- Learn how to use Virtex-4 in your specific application
- Learn about the key features of the new Xilinx ISE 6.3i integrated software environment

For your convenience, the seminar can take place at your location at a time of your choosing.

Speedway Seminar Registration -
www.em.avnet.com/v4speedway

Kit Information and Purchases -
www.em.avnet.com/virtex4lx

Ready.

Set.

Go to market.™



Xilinx® Virtex-4™ SpeedWay Seminar™

This seminar will explore the following topics: integrated PowerPC™ processors, the world's most popular embedded processor architecture, next generation Xtreme™ DSP technology, Advanced Silicon Modular Block (ASMBL) architecture and RocketIO™ serial transceivers.

Part Number	Description	Special Pricing for Seminar Attendees*
ADS-SPDWY-V4-INTRO	Xilinx Virtex-4 SpeedWay Seminar	FREE
ADS-XLX-V4LX-EVL25	Xilinx Virtex-4 LX25 Evaluation Kit	\$299.00 USD*
ADS-BASEX-BUNDLE	Xilinx Virtex-4 LX25 Evaluation Kit bundled with ISE BaseX (only available with purchase of Virtex-4 LX25 Evaluation Kit)	\$550.00 USD*
ADS-FOUNDATION-BUNDLE	Xilinx Virtex-4 LX25 Evaluation Kit bundled with ISE Foundation (only available with purchase of Virtex-4 LX25 Evaluation Kit)	\$2,400.00 USD*

*Pricing valid only within 60 days of attending a seminar.



Enabling success from the center of technology™



1 800 332 8638
www.em.avnet.com

Virtex-4 ML401 Evaluation Platform



Features

- Support for multiple clock sources and differential clock inputs
- Memory interfaces for DDR SDRAM, ZBT SRAM, and Linear Flash
- Multiple FPGA configuration modes: Platform Flash, System ACE™ CF solution, Linear Flash, and Parallel Cable-IV
- Audio and video interfaces
- Multiple user interfaces: dual PS/2, IIC Bus, RS-232, USB, and tri-mode Ethernet
- High-speed expansion module interface supporting single-ended and LVDS I/O standards
- Reference designs and IP cores for numerous applications speed up your design cycle
- A comprehensive suite of application notes guides you every step of the way
- Demonstrations ship in Platform Flash, Linear Flash, and System ACE CF solution

The Virtex-4 ML401 evaluation platform is a low-cost, full-featured development system.

Shrinking budgets and design cycles make evaluating, designing, and testing complex systems more challenging than ever before. Xilinx® provides the answer with the Virtex-4™ ML401 evaluation platform.

Powered by the XC4VLX25 device and incorporating industry-standard peripherals, connectors, and interfaces, the Virtex-4 ML401 evaluation platform provides a rich feature set that spans a wide range of applications.

Xilinx also provides expert guidance to designers with hardware-verified reference designs, application notes, and user-friendly tools.

The Virtex-4 ML401 evaluation platform specifications include:

• Xilinx devices

- XC4VLX25-FF668-10C, XC95144XL, XCCACE (System ACE CF solution), XCF32P (Platform Flash)

• Clocks

- 100 MHz oscillator, extra clock socket

• Memory

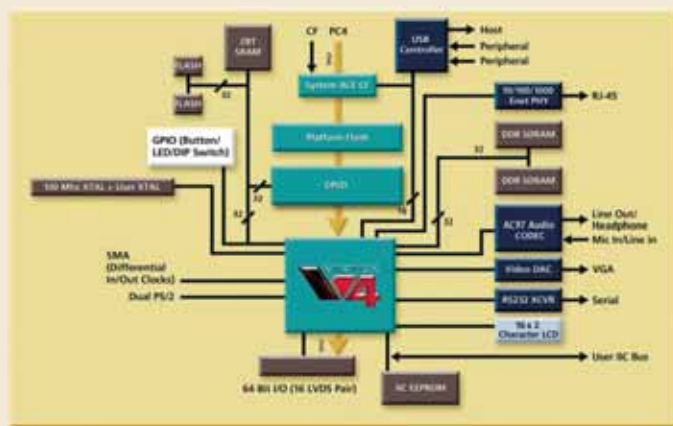
- 64 MB DDR SDRAM, 1 MB ZBT SRAM, 32 MB Compact Flash, 8 MB Flash, 4 kb IIC EEPROM, 32 Mb Platform Flash

• Display

- 16 x 2-character LCD

• Connectors and Interfaces

- Four SMA connectors (differential clocks), two PS/2 connectors (keyboard/mouse), LVDS personality module, audio (line in, line out, microphone, headphone), RS-232 serial port, USB (one host and two peripheral), Parallel Cable-IV header, DB15 VGA display, RJ-45 Ethernet port



Order your Virtex-4 ML401 evaluation platform today to get a head start on your design. For more information about the Virtex-4 FPGA family, visit www.xilinx.com/virtex4/.

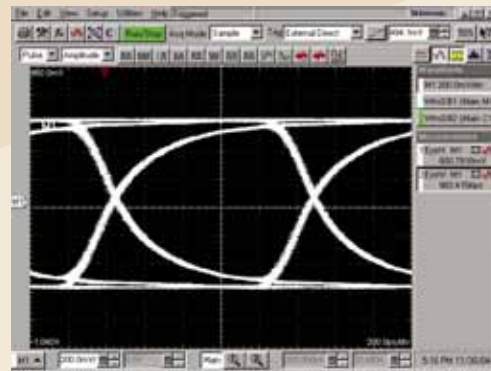


Virtex-4 FPGA Source-Synchronous Interfaces Tool Kit



Features

- Design with major differential I/O standards in networking, computing, storage, and wireless
- Pre-engineered IP and reference designs
- A unique built-in silicon feature enables 1 Gbps performance



Achieve faster, easier implementation with source-synchronous interfaces.

Today's telecom and networking systems use high-bandwidth interfaces based on LVDS, HyperTransport™, and other differential I/O standards. These standards simplify system design by lowering pin count and power consumption and improving signal integrity.

Protocols based on these standards, such as SPI-4.2, RapidIO™, and HyperTransport, are central to leading-edge system design.

Xilinx® Virtex-4™ FPGAs offer up to 1 Gbps SelectIO™ parallel I/O, with the flexibility to use any I/O pair as differential I/O. Additional benefits for higher level protocol implementation include:

- ChipSync™ source-synchronous I/O technology for dynamic precision phase alignment and data centering with per-bit de-skew
- Bitflip module supports training patterns
- Internal SerDes modules and regional clocks enable 1 Gbps DDR bandwidth

The Virtex-4 FPGA source-synchronous interfaces tool kit comes with the following Xilinx Productivity Advantage (XPA) options:

- ML450 platform, including Compact Flash, clock modules, documentation, reference designs, cables, and evaluation software
- ISE™ Foundation™ software
- IP cores: SPI-4.2, RapidIO, and GFP
- Training, Premium, and Titanium Services
- Check with your Xilinx sales representative for availability

Buy the source-synchronous interfaces tool kit today to get started on your design. For more information about the kit, the Virtex-4 FPGA family, ChipSync technology, and available optional IP, visit www.xilinx.com/virtex4/.

ML461 – Advanced Memory Development System



Features

- Memory interfaces: DDR2 SDRAM, DDR SDRAM, QDR II SRAM, RDRAM II, FCRAM II (Figure 1)
- Four Xilinx® Virtex4™ LX-25 devices
- JTAG interface
- System ACE™ Compact Flash card
- CD-ROM with complete documentation
- 5V power supply

Parameter	DDR2 SDRAM	DDR SDRAM	QDR II	RDRAM II	FCRAM II
Data Rate	534 Mbps	400 Mbps	1.2 Gbps	600 Mbps	600 Mbps
CLK Rate	267 MHz	200 MHz	300 MHz	300 MHz	300 MHz
Data Width	144-bit (DIMM) 28-bit	144-bit (DIMM) 28-bit	(72+72)-bit	36-bit	36-bit
I/O Standard	SSTL 18	SSTL 2	HSTL	HSTL	SSTL 18

Figure 1 – Memory architectures supported by ML461

Virtex-4 FPGAs make complete memory interface solutions possible.

Building interfaces to high-performance memory devices presents challenges such as high-speed synchronous data capturing, along with implementing complex physical-layer interfaces and control logic.

Virtex-4 FPGAs solve these challenges with advanced silicon capabilities, including ChipSync™ source-synchronous technology, Xesium clocking, and Smart RAM.

- ChipSync technology provides 80 ps resolution for clock-to-data alignment, ensuring reliable data capture
- 500 MHz Xesium differential global clocks minimize skew and jitter, providing increased design margins
- 500 MHz Smart RAM blocks have built-in FIFO functionality, minimizing design size
- Column-based I/O eliminates memory interface placement restrictions, alleviating board congestion

To shorten design time, Xilinx provides expert guidance in the form of free hardware-verified reference designs, application notes, user-friendly tools, and advanced development systems. This combination of unique silicon capabilities and comprehensive support enables you to build and verify robust memory interfaces quickly and easily.

The advanced memory development system, ML 461, offers

an excellent platform to develop and verify high-performance memory interfaces.

Xilinx also offers a menu-based tool, the memory interface generator, to further customize reference designs (Figure 2). The tool generates the pin placement file and a complete modular set of HDL files.

Figure 2 – Memory interface generator



You can download the reference design, application notes, memory interface generator, and other resources for memory interface designs by visiting www.xilinx.com/virtex4/. If you are interested in purchasing the ML461, please contact your local sales representative, or e-mail designkits@xilinx.com.



Memec Virtex-4 Board Solutions



Virtex-4 LC Development Kit Features

- XC4VLX25-10SF363 FPGA
- 10/100 Ethernet PHY
- 32M x 16 DDR memory
- P160 interface
- 2 x 16-character LCD
- RS232
- System ACE™ interface
- Low cost

The Virtex-4 LC development kit accelerates design time.

The Memec™ LC development kit for Xilinx® Virtex-4™ devices creates an easy-to-use yet effective Virtex-4 prototyping environment. The LC board provides prototype features common to most designers' needs, with a focus on usability in real-world applications.

The kit bundles a full-featured, expandable Virtex-4-based system board with a power supply, user guide, and reference designs. Optional Xilinx ISE™ software, JTAG cable, and application-specific P160 expansion modules are also available.



Virtex-4 MB Development Kit Features

- XC4VLX25, LX60, or SX35-10FF668 FPGA
- 10/100 Ethernet PHY
- 32M x 16 DDR memory
- 2M x 16 Flash memory
- P240 high-performance interface
- High-speed LVDS interface
- 2 x 16-character LCD
- RS232 and USB interface
- System ACE interface
- High performance

The Virtex-4 MB development kits give you maximum flexibility to target high-end applications.

The Memec MB development kits for Xilinx Virtex-4 devices provide advanced functions and interface features for your most demanding Virtex-4 prototype needs.

The MB board is available in both LX25 and LX60 densities, and for DSP applications, the SX35.

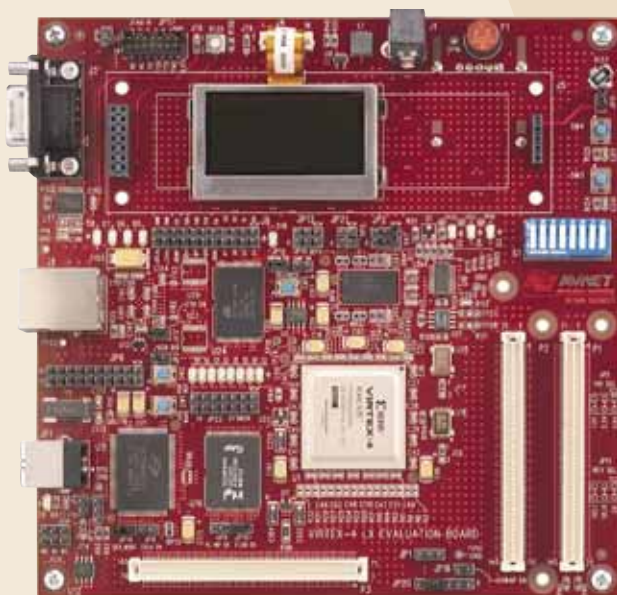
The kit bundles an expandable Virtex-4-based system board with a power supply, user guide, reference designs, and optional ISE software and JTAG cable. The new P240 expansion module standard included on the board provides both LVDS and single-ended signals to support more challenging expansion requirements.



For more information or to order your Virtex-4 development kit from Memec, visit www.memec.com/xilinx-v4/ or call (888) 488-4133 (in the U.S.) and (858) 314-8910 (outside the U.S.).



Avnet Virtex-4 Evaluation Kits



Features

- Xilinx®XC4VLX25 FF668, XC4VLX60 FF668, or XC4VSX35 FF668 FPGA
- Cypress™ CY7C68013 USB 2.0 controller
- National Semiconductor™ DP83847 10/100 Ethernet PHY
- Intel™ 8 MB Flash
- OSRAM 128 x 64 graphical display
- Micron™ 32 MB DDR SDRAM
- Texas Instruments™ CDC5801 clock multiplier/divider

Virtex-4 LX25, LX60, and SX35 Evaluation Kits are now available.

The Virtex-4 family of FPGAs delivers powerful new capabilities for designs in the programmable logic, DSP, embedded processing, and high-speed serial I/O applications domains. As a Xilinx distributor, Avnet plays a critical role in helping customers rapidly adopt the Virtex-4 solution into innovative, feature-rich end products.

Avnet is now shipping three new evaluation kits: the Virtex-4 LX25 and LX60 Evaluation Kits and the Virtex-4 SX35 Evaluation Kit (Figure 1). The LX Evaluation Kits feature an XC4VLX25 or XC4VLX60 device. These two kits are optimized for general logic integration applications.

The SX35 Evaluation Kit, which is optimized for high-performance DSP applications, uses the same board populated with a Virtex-4 XC4VSX35 device.

All three kits offer a choice of affordable, easy-to-use platforms for evaluating and experimenting with a Virtex-4 LX or SX design. And by tying in expansion cards available from Avnet, such as add-on memory, audio/video, and adapters for data conversion, these kits can serve as powerful prototyping platforms.

Purchasing any Avnet Design Kit gets you into an Avnet SpeedWay Design Workshop™ for free, where you'll learn how to leverage Xilinx solutions using real-world design examples. SpeedWay Workshops are hardware-based and lab-oriented. You'll work with real hardware and development tools to build actual designs and leave with

an in-depth knowledge of the FPGA architecture and design methods used in the lab. For more information or to register for a SpeedWay Workshop, visit www.em.avnet.com/xlxspeedwayindex/.

Virtex-4 LX Platform		
Featured Device	Avnet Part Number	Price
XCV4LX25	ADS-XLX-V4LX-EVL25	\$349.00 USD
XCV4LX60	ADS-XLX-V4LX-EVL60	\$599.00 USD
Virtex-4 SX Platform		
Featured Device	Avnet Part Number	Price
XCV4SX35	ADS-XLX-V4SX-EVL35	\$449.00 USD
Virtex-4 FX Platform		
...coming soon		

Avnet's design kits and technical workshops are powerful tools that you can leverage to increase your design advantage when implementing Virtex-4-based solutions. For more information, visit www.em.avnet.com/xlcv4kits/.



Nu Horizons Virtex-4 Development Platform



The NH401 from Nu Horizons Electronics Corp. is designed as a low-cost, high-value development platform to provide a demonstration of the Xilinx® Virtex-4™ LX/SX/FX family. The NH401 platform showcases the enormous power and flexibility of Virtex-4 FPGAs, including new and improved clock technology, system monitors, DSP blocks, Smart RAM blocks, advanced I/Os, embedded MACs, 10/100/1000 Ethernet MAC, RocketIO™ MGTs, and embedded processors (Power PC™ 405 hard-core and MicroBlaze™ soft-core processors).

The NH401 is built around a Virtex-4 FPGA and is designed to offer a user-friendly and highly useful set of features at an extremely low price point. The board is envisioned to function as an easy-to-use demonstration platform, as well as a high-performance DSP development or embedded processing platform. Included with the NH401 are simple tutorials, reference designs, and interesting demos, including a full embedded computer that can you can easily expand or adapt for your own applications.

Evaluate and implement your design by leveraging the ML401 board's rich feature set.

Feature Summary

- XC4VLX25/40/60, FX12, SX35-FF668
- Memory
 - 64 MB DDR2 SDRAM - 533 MHz
 - 1 MB ZBT SRAM
 - 32 MB CompactFlash™
 - 8 MB Flash
 - 128 kb IIC EEPROM
 - 32 Mb Platform Flash
- VGA controller (resolutions as high as 1024 x 768 at 60 Hz)
- Audio in/out CODEC (microphone in, line-in/out, and headphone output jacks)
- LCD display (16 x 2 character)
- RS232 serial port
- 2 x PS/2 (P/C keyboard and mouse)
- GPIO: 5 Buttons + 13 LEDs + 8 DIP switches
- 4 SMAs (differential clock in/out) + CLK oscillator socket
- ADC system monitor (-3V or 0-6V swing can be sampled)
- 64-bit expansion I/O connector routed for LVDS, Agilent Soft Touch connector
- 10/100/1000 Ethernet PHY
- PC4 connector (allow for JTAG debug/download via the Parallel-IV cable)
- USB host/peripheral interface
- CPLD for Flash configuration of FPGA
- High-speed frequency synthesizer - 622 MHz

Additional plug-in evaluation modules are available:

- Linear Technology high-speed A/D converters
 - 10/12/14-bit 10 to 135 Msps ADCs
- Intersil high-speed D/A interface
 - 8/10/12/14-bit 130 to 260 Msps DACs

Support for Multiple Clock Sources and Differential Clock Inputs

- Memory interfaces for DDR2 SDRAM at 533 MHz, ZBT SRAM, and Linear Flash
- Multiple FPGA configuration modes: Platform Flash, System ACE™ CF, Linear Flash, and Parallel Cable-IV
- Audio and video interfaces
- Multiple user interfaces: dual PS/2, IIC Bus, RS-232, USB, and tri-mode Ethernet
- High-speed data acquisition expansion module interface supporting single-ended and LVDS I/O standards

Optimize Your Design with Unique Built-In Silicon Features

- ChipSync™ source-synchronous technology embedded in every I/O ensures reliable data capture
- Xesium differential global clocks minimize skew and jitter for increased design margins

Finish Faster Using Proven Reference Designs

- Reference designs and IP cores for numerous applications speed up your design cycle
- A comprehensive suite of application notes guides you every step of the way

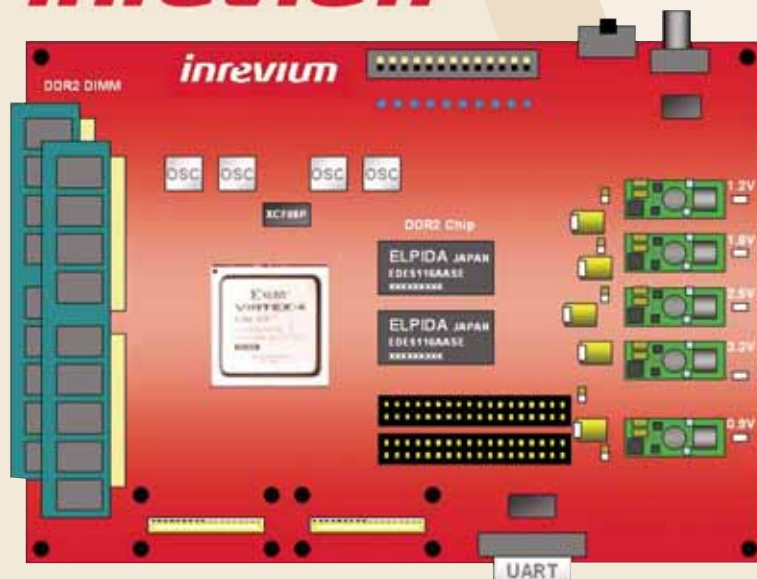
* Demonstrations ship in Platform Flash, Linear Flash, and System ACE CF formats

NU HORIZONS
ELECTRONICS CORP.

All of the designs and related documentation for the Virtex-4 board are available on the Nu Horizons website at www.nuhorizons.com/v4/.

TED DDR2 Memory Evaluation Board

inrevium



Features

- Xilinx® Virtex-4™ LX25/ LX40/ LX60 in FF668 package
- 2X DDR DIMM (533 Mbps)
- 2X DDR mounted memory (533 Mbps)
- 533 Mbps DDR2 memory controller reference design
- Board schematic/Gerber/BOM files
- Various option boards (HDL reference design)
 - DVI Tx/Rx option board
 - HDMI Tx/Rx option board
 - CameraLink I/F board
 - Optical I/F board

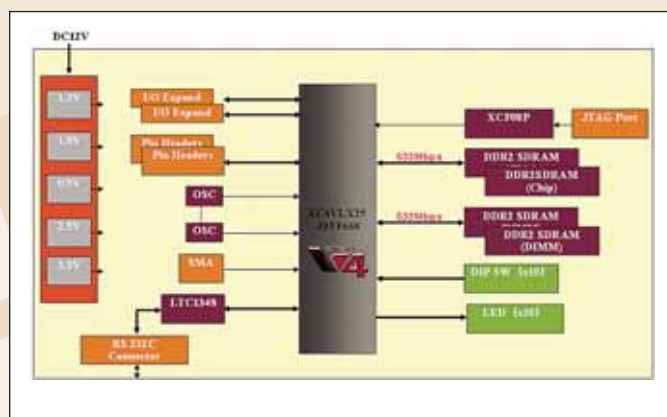
High-performance, easy-to-use, and low-cost platforms for the rapid evaluation of DDR-II memory devices.

With recent, rapid progress in memory-related technology, the standard of SDRAM is shifting from SDR to DDR, further enabling the rise of DDR2 SDRAM. It is becoming the de facto standard in the industry with its numerous advantages of low power consumption, high speed, and reduced EMI.

The TED DDR2 memory evaluation board from HiTech Global Distribution allows you to evaluate DDR2 SDRAM with the Virtex-4 LX series (LX25/40/60). The DDR2 SDRAM comprises two embedded component chips and two DIMM modules, thus allowing use in various memory evaluation applications.

Additionally, so that you can use the board immediately after purchase, the board is under plan to provide a 533 Mbps reference design.

We also offer a Gerber file as well as a board schematic file, which can assist you in developing high-speed interfaces for DDR2 SDRAM and FPGAs.



The designs and related documentation for this board are available on the HiTech Global Distribution, LLC website at www.hitechglobal.com/ted/virtex4ddr.htm.

High Efficiency, 0.6A to 2A DC-DC Buck Regulators

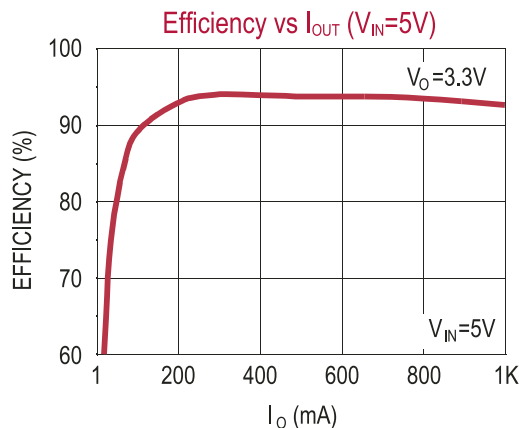
Intersil Power Management Solutions



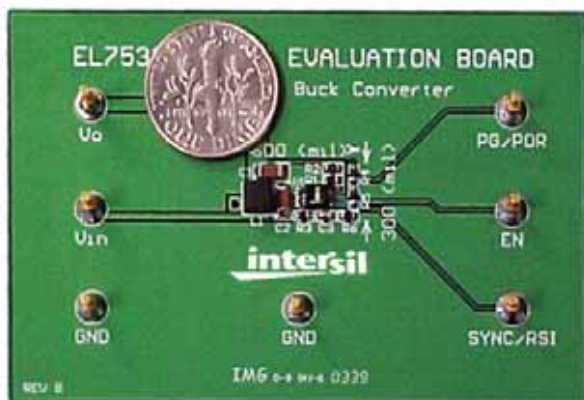
0.6A to 2A Family of Integrated FET Buck Regulators

The EL7530, EL7531, EL7532, EL7534 and EL7536 family of DC-DC buck regulators with integrated MOSFETs are simple to use, compact and full-featured. Their high efficiency makes them especially well suited for battery-operated products.

The EL7530/EL7531 devices include pulse frequency mode (PFM) and pulse width modulation (PWM) for high efficiency in standby or at full load.



EL7536 High Efficiency 1A Integrated FET Regulator



EL753X Evaluation Board with 300 mils x 600 mils Footprint

Features

- Tiny 0.97cm² total BOM footprint
- Extremely small 0.6A - 2A synchronous buck regulators
- $V_{IN} = 2.5V - 5.5V$
- 95% maximum efficiency
- 100% duty cycle (V_O close to V_{IN})
- 1.4MHz fixed PWM
 - Small passives
- PFM/PWM auto-switchable available in EL7530 and EL7531
 - 120 μ A quiescent current
- Power good signal for EL7530 and EL7531
- Power-on-reset for EL7532, EL7534 and EL7536
- External frequency synchronizable (EL7532, EL7534 and EL7536)
- Internal soft start

Applications

- Core Power Supply
- Communications Equipment
- Storage Systems
- WLAN
- Pocket PC
- Wireless Web Browsers
- GPS Navigators
- Digital Cameras
- Barcode Scanners
- Portable Instruments
- Language Translators

Learn more and get samples at <http://www.intersil.com/EL753X>

Get more technical info on Intersil's complete portfolio of High Performance Analog Solutions at

www.intersil.com/info

intersil[®]
HIGH PERFORMANCE ANALOG

©2004 Intersil Americas Inc. All rights reserved. The following are trademarks of Intersil Americas Inc.: Intersil, Intersil logo, "i" and Design.

Amplifiers | Converters | DataComm | DCPs/DCCs | Display | Interface | **Power Management** | Switch/MUX | Timing | Video

Product Selection Matrix

		Virtex-4 LX (Logic)								Virtex-4 SX (Signal Processing)				Virtex-4 FX (Embedded Processing & Serial Connectivity)					
		4VLX15	4VLX25	4VLX40	4VLX60	4VLX80	4VLX100	4VLX160	4VLX200	4V5X25	4V5X35	4V5X55	4VFX12	4VFX20	4VFX40	4VFX60	4VFX100	4VFX140	
EasyPath™ Solutions		—	—	XCE4VLX40	XCE4VLX60	XCE4VLX80	XCE4VLX100	XCE4VLX160	XCE4VLX200	—	—	XCE4V5X55	—	—	XCE4VFX40	XCE4VFX60	XCE4VFX100	XCE4VFX140	
Logic Cells		13,824	24,192	41,472	59,904	80,640	110,592	152,064	200,448	23,040	34,560	55,296	12,312	19,224	41,904	56,880	94,896	142,128	
Total Block RAM (kbits)		864	1,296	1,728	2,880	3,600	4,320	5,184	6,048	2,304	3,456	5,760	648	1,224	2,592	4,176	6,768	9,936	
Digital Clock Managers (DCM)		4	8	8	8	12	12	12	12	4	8	8	4	4	8	12	12	20	
Phase-matched Clock Dividers		0	4	4	4	8	8	8	8	0	4	4	0	0	4	8	8	8	
Max SelectIO™		320	448	640	640	768	960	960	960	320	448	640	320	320	448	576	768	896	
Max Differential I/O Pairs		160	224	320	320	384	480	480	480	160	224	320	160	160	224	288	384	448	
XtremeDSP™ Slices		32	48	64	64	80	96	96	96	128	192	512	32	32	48	128	160	192	
Analog-to-Digital Converters (ADC)		0	0	0	0	1	1	1	1	0	0	0	0	0	0	1	1	1	
PowerPC™ Processor Blocks		—	—	—	—	—	—	—	—	—	—	—	1	1	2	2	2	2	
10/100/1000 Ethernet MAC Blocks		—	—	—	—	—	—	—	—	—	—	—	2	2	4	4	4	4	
RocketIO™ Serial Transceivers		—	—	—	—	—	—	—	—	—	—	—	0	8	12	16	20	24	
Configuration Memory Bits		4,875,392	8,037,312	12,647,680	18,315,520	24,101,440	31,818,624	41,863,296	50,648,448	9,651,072	14,476,608	24,088,320	5,017,088	7,641,088	15,838,464	22,262,016	35,122,240	50,900,352	
Package	Area	4VLX15	4VLX25	4VLX40	4VLX60	4VLX80	4VLX100	4VLX160	4VLX200	4V5X25	4V5X35	4V5X55	4VFX12	4VFX20	4VFX40	4VFX60	4VFX100	4VFX140	
SF363	17 x 17 mm	240	240										240	240					
FF668	27 x 27 mm	320	448	448	448					320	448		320	320					
FF676	27 x 27 mm	320	448							320	448		320	448					
FF1148	35 x 35 mm		640	640	640	768	768	768				640							
FF1513	40 x 40 mm						960	960	960										
FF672	27 x 27 mm												320 (8)¹		352 (12)¹	352 (12)¹	576 (20)¹		
FF1152	35 x 35 mm														448 (12)¹	576 (16)¹	768 (20)¹		
FF1517	40 x 40 mm																768 (20)¹	768 (24)¹	
FF1760	42.5 x 42.5 mm																	896 (24)¹	

Pb-free solutions are available. For more information about Pb-free solutions, visit www.xilinx.com/pbfree/.

1. Number of available RocketIO Multi-Gigabit Transceivers

Important: Verify all data in this document with the device data sheets found at <http://www.xilinx.com/partinfo/databook.htm>

Product Selection Matrix

		CLB Resources				Memory Resources			DSP	CLK Resources			I/O Features				Speed	PROM		
		System Gates (see note 1)	CLB Array (Row x Col)	Number of Slices	Logic Cells (see note 2)	CLB Flip-Flops	Max. Distributed RAM Bits	# Block RAM	Block RAM (bits)	Dedicated Multipliers	DCM Frequency (min/max)	# DCMs	Frequency Synthesis	Phase Shift	Digitally Controlled Impedance	Number of Differential I/O Pairs			Maximum I/O	
		Spartan-3 Family – 1.2 Volt (see note 3)																		
		XC3S50	50K	16 x 12	768	1,536	12K	4	72K	4	24/330	2	YES	YES	YES	56	124	I/O Standards Single-ended LVTT, LVCMOS3.3/5/1.8/ 1.5/2, PCI 3.3V – 2/64-bit 33MHz, SSTL2 Class I & II, SSTL1.8 Class I, HSTL Class I, III, HSTL1.8 Class II & III, GTL, GTL+ Differential LVDS2.5 Bus LVDS2.5, Ultra LVDS2.5, LVDS ext2.5, RDS, LVDT2.5, LVPECL	Commercial Speed Grades (slowest to fastest) Industrial Speed Grades (slowest to fastest)	Configuration Memory (Bits)
		XC3S200	200K	24 x 20	1,920	3,840	30K	12	216K	12	24/330	4	YES	YES	YES	76	173			
		XC3S400	400K	32 x 28	3,584	7,168	56K	16	288K	16	24/330	4	YES	YES	YES	116	264			
		XC3S1000	1000K	48 x 40	7,680	15,360	120K	24	432K	24	24/330	4	YES	YES	YES	175	391			
		XC3S1500	1500K	64 x 52	13,312	26,624	208K	32	576K	32	24/330	4	YES	YES	YES	221	487			
		XC3S2000	2000K	80 x 64	20,480	40,960	320K	40	720K	40	24/330	4	YES	YES	YES	270	565			
		XC3S4000	4000K	96 x 72	27,648	55,296	432K	96	1,728K	96	24/330	4	YES	YES	YES	312	712			
XC3S5000	5000K	104 x 80	33,280	66,560	520K	104	1,872K	104	24/330	4	YES	YES	YES	344	784					

Note: 1. System Gates include 20-30% of CLBs used as RAMs
2. For Spartan-3, a Logic Cell is defined as a 4-input LUT + flip-flop
3. Automotive Q-Grade Solutions for Spartan-3 will be available 2H2004.

Important: Verify all data in this document with the device data sheets found at <http://www.xilinx.com/partinfo/databook.htm>

Package Options and User I/O¹

Spartan-3 (1.2V)										
Pins	Area ²	I/O's	XC3S50	XC3S200	XC3S400	XC3S1000	XC3S1500	XC3S2000	XC3S4000	XC3S5000
PQFP Packages (PQ) – wire-bond plastic QFP (0.5mm lead spacing)										
208	30.6 x 30.6 mm	124	141	141						
VQFP Packages (VQ) – very thin TQFP (0.5mm lead spacing)										
100	16.0 x 16.0 mm	63	63							
TQFP Packages (TQ) – thin QFP (0.5mm lead spacing)										
144	22.0 x 22.0 mm	97	97	97						
FGA Packages (FT) – wire-bond fine-pitch thin BGA (1.0 mm ball spacing)										
256	17 x 17 mm		173	173	173					
FGA Packages (FG) – wire-bond fine-pitch BGA (1.0 mm ball spacing)										
320	19 x 19 mm				221	221	221			
456	23 x 23 mm				264	333	333			
676	27 x 27 mm						391	487	489	
900	31 x 31 mm								565	633
1156	35 x 35 mm								712	784

Note 1: Numbers in table indicate maximum number of user I/Os
Note 2: Area dimensions for lead-frame products are inclusive of the leads.
Pb-free solutions are available. For more information about Pb-free solutions visit www.xilinx.com/pbfree/.

For the latest information and product specifications on all Xilinx products, please visit the following links:

FPGA and CPLD Devices
<http://www.xilinx.com/devices/>

Software
<http://www.xilinx.com/isel>

IP Reference
<http://www.xilinx.com/ipcenter/>

Configuration and Storage Systems
<http://www.xilinx.com/configsolns/>

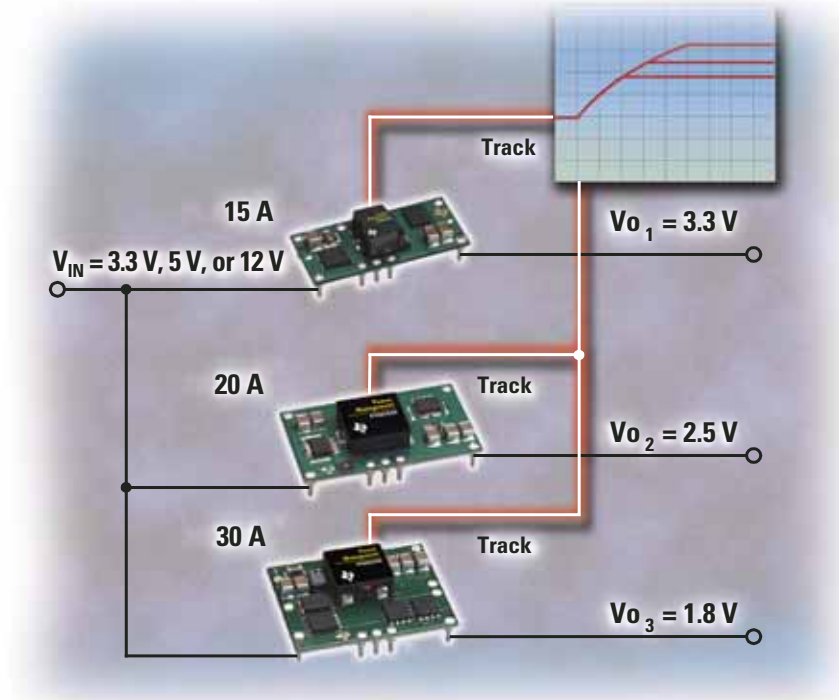
Development Reference Boards
http://www.xilinx.com/board_search/

Global Services
<http://www.xilinx.com/support/gsd/>

Packaging
<http://www.xilinx.com/packaging/>

The Industry's Most Advanced Plug-In Power Modules

Featuring TI's New Auto-Track™ Sequencing



► Applications

- Networking
- Servers
- Data communications
- Workstations
- Industrial electronics

► Features

- Auto-Track sequencing simplifies power up/down sequencing of multiple modules
- Pre-bias startup capability allows use with all ASICs and FPGAs
- Margin up/down provides for additional test capability during manufacturing
- A 96% efficiency rating means more power in a smaller package
- Point-of-Load Alliance (POLA) compatibility assures interoperable second sources

The new **PTHxx** family of plug-in power modules from Texas Instruments provides industry-leading features that allow designers to take charge of point-of-load (POL) power problems and designs. New Auto-Track sequencing via single-pin control simplifies multimodule power up/down. In addition to those listed below, other key features include wide adjustable output voltage, on/off inhibit, overcurrent protection and remote sense.

Samples shipped in 24 hours.

Series	Input Bus (V)	I _{OUT} (A)	Auto-Track Sequencing	Pre-bias Startup	Margin Up/Down	Thermal Shutdown
PTH03050/5050	3.3/5	6	✓	✓		
PTH12050	12	6	✓	✓		
PTH03060/5060	3.3/5	10	✓	✓	✓	
PTH12060	12	8	✓	✓	✓	
PTH03010/5010	3.3/5	15	✓	✓	✓	
PTH12010	12	10	✓	✓	✓	
PTH03020/5020	3.3/5	20	✓	✓	✓	✓
PTH12020	12	16	✓	✓	✓	✓
PTH03030/5030	3.3/5	30	✓	✓	✓	✓
PTH12030	12	20	✓	✓	✓	✓

POLA™



FREE!
Plug-in Power and
Power Management
Selection Guides

► www.ti.com/xcell
1-800-477-8924, ext. 1202

Datasheets, Samples,
Plug-in Power and
Power Management
Selection Guides

Technology for Innovators™

 **TEXAS INSTRUMENTS**

The Right Power Supply. The Right Partners.



Optimizing your FPGA power requirements at the beginning of your design cycle eliminates last-minute surprises and delays. Xilinx industry-leading partners—Texas Instruments, National Semiconductor, Linear Technology, and Intersil—make it fast and easy with simple, comprehensive reference guides and support. It's a powerful first step in developing more robust and reliable Xilinx FPGA designs



www.xilinx.com/powercentral



ONE FAMILY.
MULTIPLE PLATFORMS.
ENDLESS POSSIBILITIES.



OPTIMIZED
FOR LOGIC



OPTIMIZED
FOR DSP



OPTIMIZED FOR
PROCESSING/CONNECTIVITY



Introducing the world's first multi-platform, domain-optimized FPGA family—delivering breakthrough capabilities and performance at every price point.

THE FREEDOM TO CHOOSE

For the first time ever, you can select from multiple FPGA platforms, optimized for application domains. You choose the exact capabilities you want. You pay only for what you need. Virtex-4 FPGAs are built upon our unique ASMBL™ (Advanced Silicon Modular Block) architecture, enabling Xilinx to *assemble* logic, memory, I/O, DSP, processors and more, giving you complete freedom of choice.

A SOLUTION FOR EVERY SYSTEM DESIGN CHALLENGE

The three Virtex-4 platforms—LX, SX, and FX—offer you up to 200,000 logic cells, and 500 MHz tuned performance. Our new ChipSync™ technology simplifies source-synchronous interfaces. You can implement serial protocols at any speed from 600 Mbps to 11.1 Gbps with RocketIO™ multi-gigabit transceivers. Hardware acceleration for the embedded PowerPC™ is easy with our auxiliary processing unit. And with XtremeDSP™ delivering 256 GMACS, you can solve those ultra-high performance DSP challenges.

All of your design possibilities just became realities. See for yourself at www.xilinx.com/virtex4.



*Easiest to Use
Software*

 **XILINX**
The Programmable Logic Company™
www.xilinx.com

LOWEST SYSTEM COST • HIGHEST SYSTEM PERFORMANCE