ISSUE 48, SPRING 2004
XCELL JOURNAL
XILINX, INC.

Issue 48
Spring 2004

Xcell journal

THE AUTHORITATIVE JOURNAL FOR PROGRAMMABLE LOGIC USERS

Xilinx Turns 20

XILINX®

Introducing the

SPARTAN™-3 Platform FPGAs.

The New Generation of *Low Cost Solutions.*

Now up to 5 Million Gates on 90nm

The world's lowest-cost FPGAs

The first devices ever manufactured in 90nm process technology, our new Spartan-3 FPGAs deliver unbeatable price and performance. Packing a ton of I/Os, with densities ranging from 50K to 5 Million gates, you've got the lowest cost per pin and lowest cost per gate in the industry—and it's all in the most cost-efficient die size available.

All the density and I/O you need . . . at the price you want

The new Spartan-3 FPGAs allow you to take your design idea from the white board to the circuit board in a huge range of applications. With cost savings in manufacturing, easy integration of soft core processors, superior DSP performance and full I/O support with total connectivity solutions, you get all you need at the price you want.

Easiest to use Software

The Spartan-3 Platform FPGAs are ready for a new generation of designs. . . are you?

MAKE IT YOUR ASIC

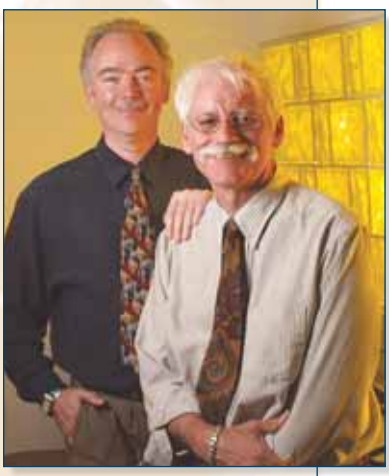
XILINX®

The Programmable Logic Company℠

For more information visit
www.xilinx.com/spartan3

Award Winner 2003
The Year's Ten Best Web Support Sites

FORTUNE® 2003
100 BEST COMPANIES TO WORK FOR



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

MANAGING EDITOR — Tom Durkin, tom.durkin@xilinx.com, 530-271-0899

ASSOCIATE 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

Xcell journal

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

20 Years and 900 Patents Later...

Xilinx is now entering its 20th year in business, and in that time we have not only created a whole new category of devices – FPGAs – but we have also created a company like no other.

Our hardware and software technology, our business model, and our corporate culture have started a revolution in logic design. As our device performance has dramatically increased and prices have dramatically decreased, our technology has become the obvious choice for new designs. From miniature consumer devices to large switching systems, there simply is no better way to create new designs quickly and efficiently. And the trend toward higher density, lower cost, and faster development cycles will continue well into the future.

While our technology has steadily opened new markets and design possibilities, we have also built a great place to work – at last count we were *Fortune* magazine's fourth best company to work for. Our fast-paced innovation is a direct result of our corporate culture and our values – it makes a strategic difference when people enjoy their work.

We also support others in their quest for excellence. As you will see from the SETI article on page 8, Xilinx has long been a part of this far-reaching research program.

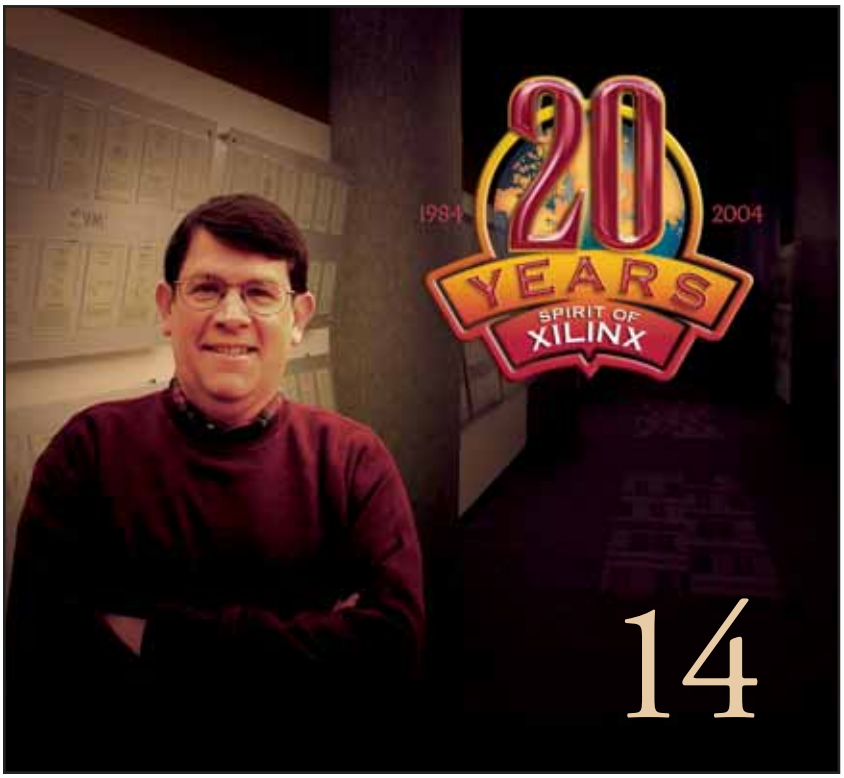
What's Next?

Our new Virtex-4™ FPGA family has just been announced, with a revolutionary modular architecture. Our new Application Specific Modular Block architecture (ASMBL) segments function blocks into interchangeable columns, rather than the traditional squares on a grid. Combined with flip-chip technology (which allows us to make I/O interconnects anywhere on the device – not just the periphery) this architecture allows us to provide a range of new devices optimized for specific classes of applications. Once again, you get even higher performance and lower costs. Plus, you will have the ability to select just the right mix of features and functions, which makes your design process easier. You'll soon be seeing a lot more about the Virtex-4 family.

In the last 19 years, we have been granted more than 900 patents; that comes out to about a patent a week – not a bad record of innovation for any company. May the next 20 years be just as much fun.

Carlis Collins

Carlis Collins
Editor-in-Chief



Celebrating 20 Years of Innovation

As Xilinx marks its 20th anniversary, a Xilinx Fellow recalls the birth of programmable technology.

Possibilities Not Yet Imagined

Programmable logic is moving from a niche technology to a mainstream market segment.

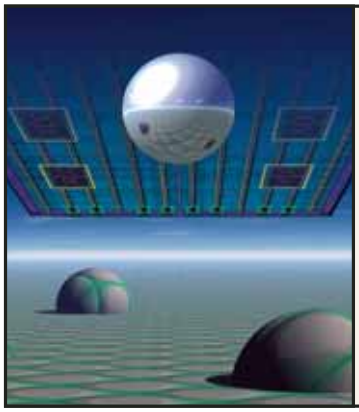
COVER STORY

SETI Researchers Sift Interstellar Static for Signs of Life

The raw computational power of Xilinx FPGAs in ganged arrays drives the international search for extraterrestrial intelligence at 10^{15} ops per second.

SPRING 2004, ISSUE 48

Xcell journal



Emulate 8051 Microprocessor in PicoBlaze IP Core

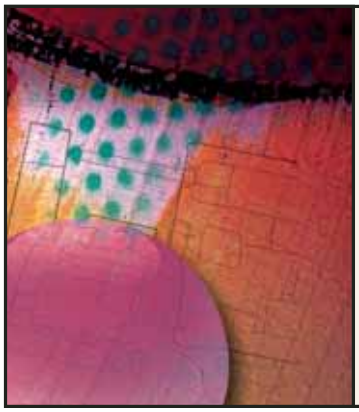
Put the functions of a legacy microprocessor into a Xilinx FPGA.

18

Improve Your Productivity with the Embedded Development Kit v6.1i

The new version of EDK offers automatic generation of complex implementation details, new simulation capabilities, and new IP cores.

24



Unleash Your Creativity with Embedded Linux on Virtex-II Pro FPGAs

Xilinx has partnered with MontaVista Software to provide a customized embedded Linux solution for Virtex-II Pro FPGAs.

38

Put the Right Bus in Your Car

The amazing array of features available in today's cars has spawned new in-vehicle bus standards.

62

Possibilities Not Yet Imagined

As we enter our 20th year, programmable logic is moving from a niche technology to a mainstream market segment.

by Wim Roelandts
CEO, Xilinx, Inc.

Programmable logic technology is advancing at a phenomenal rate. Our FPGAs now contain advanced processors, gigabit-per-second transceivers, vast numbers of logic cells, and advanced features that were unheard of just two short years ago. Plus, we've lowered our pricing significantly across the board, due to our use of the most advanced 90 nm processes and other leading manufacturing technologies. We have progressed far in a very short time – and there's more on the way.

During the last 20 years, Xilinx has grown from a small startup company to having more than a billion dollars in revenue. Each step of the way, we've had to transform ourselves to meet the demands of a growing organization. Many companies fail because they cannot manage this constant transformation.

Today, we live in "interesting times" because there are significant changes coming over the next few years.

Grand Changes

Today, programmable logic is 20 percent of the total logic market. I believe that we can get to 50-60 percent somewhere in the future, and that again requires us to change our approach. Because to be a mainstream company, we have to be a low-cost producer; we have to serve a broad range of markets. Historically we've served a relatively low-volume, high-cost, narrow market.

You can see that now we have a global transformation in front of us, a transformation that requires us to discover and learn and sell into new, high-volume markets. We must also learn how to sell into new, more strategic applications. We must sell not only to engineers, but to the systems architects and vice presidents, who care most about the long-range implications of today's technology.

At the same time, our international business is growing very rapidly. Asia, which in 2000 was less than 15 percent of our revenue, is now more than 35 percent of our revenue, which again forces us to change the way we do things, from engineering to support.

I believe that the next big cycle of innovation ahead of us is going to be the convergence of digital consumer electronics. For the last 20 years, innovation has been driven by the PC market. I believe that the next 20 years is going to be driven by consumer electronics, both for the home and for the car, and I think that the center of that is going to be in Asia.

Product Changes

We are also moving into new markets, with new processors and very high-speed transceivers capable of managing tremendous amounts of data. In many designs, there is nothing faster than our programmable logic devices.

Historically, when programmable logic was mainly used as glue logic, the decision to use programmable logic products was made late in the design cycle, typically less than a year before the product went into production. Therefore, we got involved late in the design cycle. Today, because FPGAs are the primary component in many systems, we have to sell early in the design cycle – two to three years before the product goes into production. This requires a very different approach to sales and marketing.

For example, we have just released our Virtex-II Pro™ X FPGA, which is another important milestone. It will be the first time that a 10 gigabit transceiver is integrated with an FPGA.

Our next major product introduction is our Virtex™-4 FPGA family. It uses very aggressive seamless technology and is a completely new design concept. Every single function was redesigned, changed, modified, and significantly improved to both increase performance and reduce costs.

The most innovative part of our new Virtex-4 family is the physical layout. We have moved away from the traditional layout (with the chip in the center and the I/O on the periphery) toward the use of flip-chip technology, which allows us to have a very new structured columnar architecture. So, we put different resources in strips or columns of technology and then stitch the columns together.

With flip-chip packaging, we can connect I/Os to any part of the chip, not just the periphery. This increases performance, decreases noise, and makes it far easier to implement efficient designs. The Virtex-4 FPGA design is truly revolutionary.

The Culture and Spirit of Xilinx

Our products and services are often imitated by our competition, but our corporate culture of innovation is very difficult to match – our culture is the most important long-term competitive advantage we have.

Culture is not something that is fixed; it must change and grow with circumstances; it must keep evolving and changing. Our culture itself must not only change as needed, it must support change and innovation on all levels of our company, from engineering and support to marketing and sales. Innovation is alive and well at all levels of the Xilinx culture.

Partnerships are a key part of our culture as well. Xilinx is extremely strong in creating win-win solutions with our partners, because the way we treat our partners is very much how we treat our own employees. We make sure they get something more out of the relationship than just business from Xilinx. Because for me, our business partners are part of the Xilinx ecosystem (just like our customers), and are necessary for the health and growth of the whole.

We do interesting work at Xilinx, and that's what motivates us. Our employees consistently report in surveys that their work is meaningful and that they are proud of their work. That's why we focus on industry leadership – because if you do something that is really top-notch, really state-of-the-art, then clearly your work has value and meaning. It's not just leadership as a way of getting market share or being number one in the market. Our leadership is also a very strong motivator for our people. Ultimately, that's a key advantage that cannot be matched by any competitor.

In companies like Xilinx, people jump in and help out when needed – it's a very important part of being an innovative company. There's no finger pointing and no blaming here, just saying "hey, let's go forward, let's help out." All of this is part of the Spirit of Xilinx.

I believe you can treat people with respect and still be a leader in your industry. And the difference is innovation. Because if you innovate, there's no competition. If you have the most innovative product and unique features, there is no competition. And when there is no competition, you don't have to drive people to work 12 hours a day.

In fact, I believe that if you really want to have innovation you cannot drive people too hard, because innovation occurs when people are not too busy. If you look at our fundamental core culture, it's innovation; innovation in everything we do. And that is also in line with the high-tech industry, where things keep changing continuously.

Now let's not forget that there is something else that is needed, and that is business success. Whatever you do, if it doesn't create business success, it's not going to work. Fortunately, our success is tremendous. A couple of years ago we were not even the number one company in programmable logic, but we have come back with a vengeance – we now have 50 percent market share, gaining almost 20 points of market share in the last six years or so – which is really remarkable.

Conclusion

Xilinx is strong because our programmable logic technology is unmatched, and it is the right technology for the future of logic design. There is no better way to design and develop next-generation equipment. We are also strong because our culture supports continued innovation at all levels of our company. It's a competitive advantage that cannot easily be surpassed.

I think that we are just at the beginning of what is possible. With our enabling technology and our customers' creativity, I'm sure that there are going to be new ways of doing things that will emerge; possibilities we have not yet imagined. Σ

SETI Researchers Sift Interstellar Static for Signs of Life

The raw computational power of Xilinx FPGAs in ganged arrays drives the international search for extraterrestrial intelligence at 10^{15} ops per second.

by Tom Durkin
Managing Editor, *Xcell Journal*
Xilinx, Inc.
tom.durkin@xilinx.com

You don't have to leave Earth to find intelligent life on other worlds. All you have to do is tune in ... at the right time ... on the right frequency ... in the right direction ... with the right spectrometer ... using the most powerful supercomputer on this planet.

With the support of the Xilinx University Program (XUP), the University of California at Berkeley has emerged as the world leader in the search for "ET" [pronounced EE-tee, as in the popular fantasy motion picture *E.T.*]. UC Berkeley operates about a half-dozen different SETI (Search for Extraterrestrial Intelligence) projects under the umbrella of the SERENDIP (Search for Extraterrestrial Radio Emissions from Nearby Developed Intelligent Populations) Program.

"When we started 25 years ago, we built a machine that could listen to a hundred channels at once. We thought that was amazing," says Dan Werthimer, Ph.D., director of the SERENDIP SETI Program at UC Berkeley. "That was called SERENDIP I. Then we went to 65,000 different channels with SERENDIP II – and then Xilinx technology allowed us to go to four million channels with SERENDIP III." And in 1997, "We went to 168 million channels with SERENDIP IV."

Later this year, SERENDIP V will go online at the Arecibo Observatory in Puerto Rico with the capability of simultaneously processing data from five billion channels (Figure 1), using several hundred Virtex™-II XC2V6000 and XC2V1000 platform FPGAs populating dozens of racks of spectrum analyzer boards.

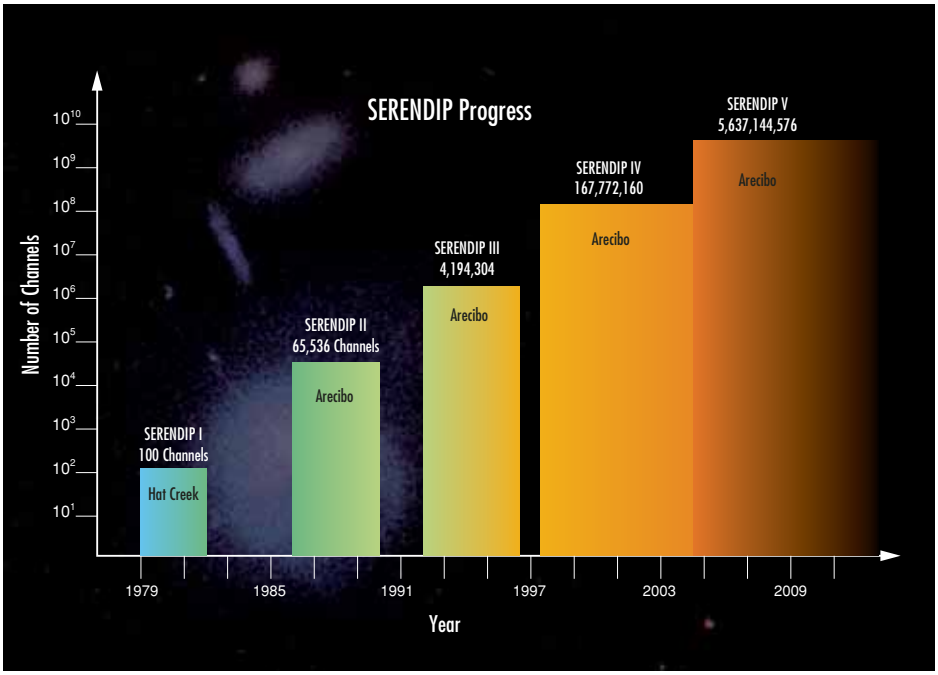


Figure 1 – SERENDIP DSP growth follows Moore's Law

With such an awesome capability to collect massive amounts of data, the SERENDIP scientists need far more computing power than they could possibly have with the high-end Sun Microsystems™ workstations at the Berkeley Space Sciences Laboratory. That's where you and I come in. SETI@home volunteers comprise the largest supercomputer on the planet.

Meanwhile, in the next few years, UC Berkeley's Radio Astronomy Laboratory and the SETI Institute of Mountain View, Calif., will build the Allen Telescope Array, a bold innovation in radio telescope design – and a powerful new tool for SETI research. As with SERENDIP, Xilinx will provide the core technology to enable real-time digital signal processing (DSP) at the unprecedented speed of 10^{15} ops per second.

And what happens if we do find ET? Virtually all of the world's scientific community of SETI researchers have endorsed a United Nations treaty that requires the free and open disclosure of everything discovered and deciphered. How the rest of humanity will react is anybody's guess. A lot of it depends, says Werthimer, on whether the ET signal is accidental or intentional.

Tuning In to ET

Radio waves (including television, radar, cell phones, and other microwave telecommunications) are considered the optimum band of the electromagnetic spectrum for interstellar communication. Radio wavelengths are relatively free of the absorption and noise that afflict other areas of the spectrum. Additionally, stars are generally quiet in the radio wavelengths. This makes radio frequencies a natural candidate for intentional interstellar communications – or "leakage" of local transmissions.

Just as the "local transmissions" of American television shows, such as "I Love Lucy" and "The Honeymooners," leaked out into space 50 years ago (and now have passed thousands of star systems), it is conceivable that we could someday intercept an extraterrestrial situation comedy show.

Anatomy of Arecibo

Operated by Cornell University and the National Science Foundation, the National Astronomy and Ionosphere Center Arecibo Observatory is the largest radio telescope on this planet (Figure 2).

The spherical reflector dish measures 1,000 feet (305 meters) across and covers 20 acres. In what is considered a valid scientific calculation, Werthimer says the dish could theoretically hold 10 billion bowls of cornflakes. Milk, however, would quickly drain out the almost 40,000 perforated aluminum panels that make up the dish.

Suspended 450 feet (137 meters) above the dish is a 900-ton (816 metric tons) platform that can be placed with millimeter precision anywhere up to 20 degrees from the vertical. A "Gregorian dome" on the platform contains two subreflectors (secondary and tertiary) to further focus deep space radio emissions.

The platform also houses ultra-sensitive radio receivers cooled with liquid helium (to reduce electron noise) so the infinitesimally weak signals from outer space can be picked up amidst all the interstellar static and radio interference generated on Earth, orbiting satellites, and probes launched from Earth.

Figure 2 – Arecibo Observatory, Puerto Rico

"... the name of the game in SETI is to search through as many frequencies as possible." – Aaron Parsons, SERENDIP V design engineer

Piggyback SETI

While most radio astronomers are lucky to get a day or two a year to use the Arecibo Observatory, "We figured out how to use the telescope 24 hours a day all year round by having our own feed antenna," Werthimer says with a certain amount of glee.

"The problem with that is that we don't get to point the telescope, but that's okay, because we don't know where to look anyway," he grins.

"We call it piggyback SETI."

Spectacular Spectrometers

Although radio telescope antennas are visually impressive and quite essential, they are useless without the instruments that receive and process the signals. The real guts of radio telescopes are spectrometers, or spectrum analyzers, such as SERENDIP IV.

SERENDIP IV

The SERENDIP IV spectrometer at Arecibo consists of 120 Xilinx FPGAs on 40 spectrum analyzer boards working in parallel to scan 168 million narrow-band (0.6 Hz) channels every 1.7 seconds.

Each SERENDIP IV board computes a four million point Fast Fourier Transform (FFT). This four million point FFT is broken down into three smaller FFTs (128, 128, and 256 points each). Xilinx chips comb the resulting power spectra for strong narrow-band signals and report their findings to the back-end computers at Berkeley for subsequent analysis, Werthimer says.

SERENDIP V

"We don't know what frequency ET will be transmitting at, so the name of the game in SETI is to search through as many frequencies as possible," explains Aaron Parsons, a design engineer at the Berkeley Space Sciences Laboratory. Parsons is designing SERENDIP V, the next-generation spectrometer that will be able to process five billion channels simultaneously.

As of press time, SERENDIP V was still on the drawing board, but it is on schedule to be installed at Arecibo later this year.

The whole spectrometer will consist of 40 spectrum analyzer boards, each performing a pair of 64 million point FFTs to handle a real-time signal bandwidth of 100 MHz, Parsons says. Because Xilinx platform FPGAs have the capability of interfacing with double data rate DRAM memory chips, SETI engineers will be able to fit this 64 million point FFT onto a single Virtex-II XC2V6000 FPGA.

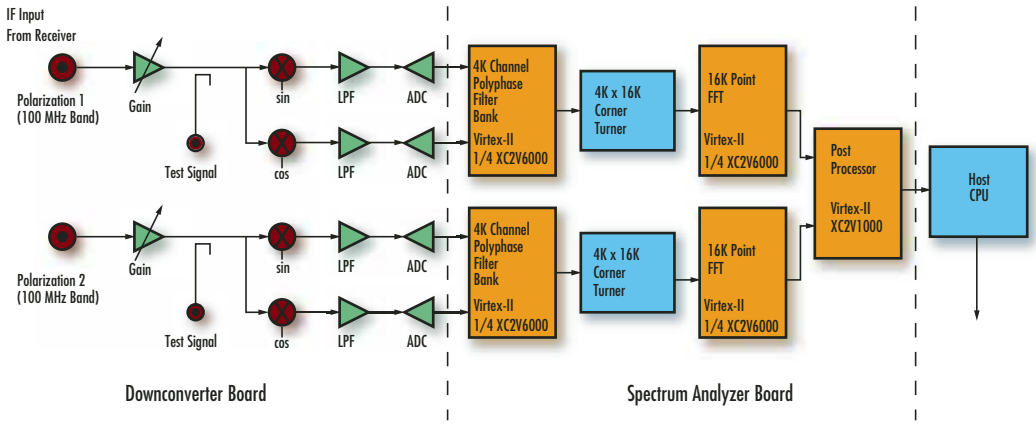


Figure 3 – SERENDIP V spectrometer module

Spectrum Splitting

"We did this by first cutting the spectrum into coarse frequency bins using the characteristic frequency response of a 4,096-channel polyphase filter bank," Parsons explains (Figure 3). "The output data were then re-ordered using 256 MB of DRAM and broken into 16,384 smaller bins using a dual flow-through FFT we developed. It uses one-fourth of the space of a traditional FFT, he says with pride.

Finally, information about the best signals will be passed to a CPU over a compact PCI backplane using a Virtex-II XC2V1000 running a Xilinx PCI core, according to Parsons.

Xilinx Chief DSP Architect Chris Dick, Ph.D., consulted on the design of SERENDIP V. "The signal processing requirements in the SETI program present significant computational and I/O challenges that can only be met using Xilinx FPGAs," Dick asserts. "Traditional processor-based approaches just cannot deliver the performance required for this challenging application.

"The highly parallel compute fabric and I/O capability of the FPGA, however, are well suited to support the computational requirements of the filter banks and FFTs used in the polyphase transform channelizer," Dick says.

Key aspects of SERENDIP V were realized using a recent generation design flow from Xilinx called System Generator for DSP. This visual programming development environment is based on The MathWorks Simulink® interactive tool for modeling, simulating, and analyzing dynamic, multidomain systems. It provides a natural framework for rapidly specifying and verifying complex signal processing systems, according to Dick.

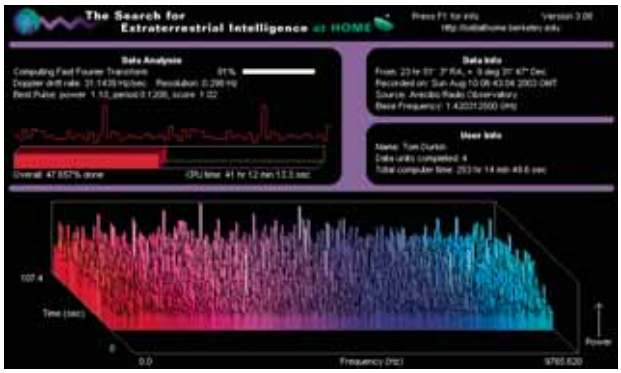


Figure 4 – SETI@home "screen saver" data analysis program

SETI@home Wants You

As fast and as efficient as "ganged" (parallel) FPGAs are, the real-time data processed by SERENDIP IV still needs much further analysis. After the DSP algorithms in SERENDIP IV break down the incoming signal into 168 million channels, some of the outgoing data is recorded onto high-density digital linear tape – about one 35 GB tape per day.

The tapes are shipped to the Berkeley Space Sciences Laboratory. Even with high-end Sun workstations, the amount of data far exceeds the lab's ability to crunch the data. Thus, the SETI@home project was born. The SERENDIP scientists decided to farm out the massive computing task to idle computers all over the Internet (distributed grid computing), so they created SETI@home "screen saver" software.

Calling the SETI@home data analyzer software a screen saver is a misnomer. The only resemblance the data analyzer has to real screen savers is that it only works when your computer is idle – and the graphical display of the data analysis in action is semi-hypnotic (Figure 4).

If you want to join the hunt for ET, you can download the free SETI@home data analysis software at *http://setiathome.berkeley.edu*. There are versions for Windows, Macintosh, Unix, Linux, BeOS, OS/2, OpenVMS, and other operating systems.

Once you've got the software loaded, the SETI@home server at Berkeley sends you a 0.34 MB "work unit." This is a very small chunk of data from the Arecibo tapes. The SETI@home data analyzer performs anywhere from between 2.4 trillion and 3.8 trillion floating point calculations – including FFTs, de-chirping, and baseline smoothing, among others.

Once the data is processed, the SETI@home program notifies you that it wants to report its results back to Berkeley and acquire another work unit. The only time the SETI@home data analyzer needs to be online is when the data is being transferred.

"When we get the data back from the participants, we comb through the strong signals looking for ET," Werthimer says.

Super Computing

The SETI@home project is an awesome display of the power of distributed grid computing. Starting with a base of 1,500 volunteers in 1998, SETI@home has grown to more than 4.7 million participants in 226 countries, with 2,000 new volunteers signing up daily, Werthimer reports.

"The SETI@home volunteers have formed the planet's largest supercomputer, averaging 60 teraflops [60 trillion floating point operations per second] and donating about 1,200 years of CPU time daily," he says. "The search for ET is truly an international effort."

Allen Telescope Array

While the SETI research will continue at Arecibo, the SETI Institute and the Radio Astronomy Laboratory at UC Berkeley have begun to build the Allen Telescope Array at the Hat Creek Observatory near Mt. Lassen in Northern California.

Underwritten by a large donation from Paul Allen, co-founder of Microsoft, the Allen Telescope Array will eventually grow to be a huge, expandable antenna farm (Figure 5) of as many as 350 20-foot (6.1-meter) offset Gregorian dishes with 8-foot (2.4-meter) secondary antennas. This innovative telescope design will cost much less than an equivalent single dish telescope, according to Werthimer.

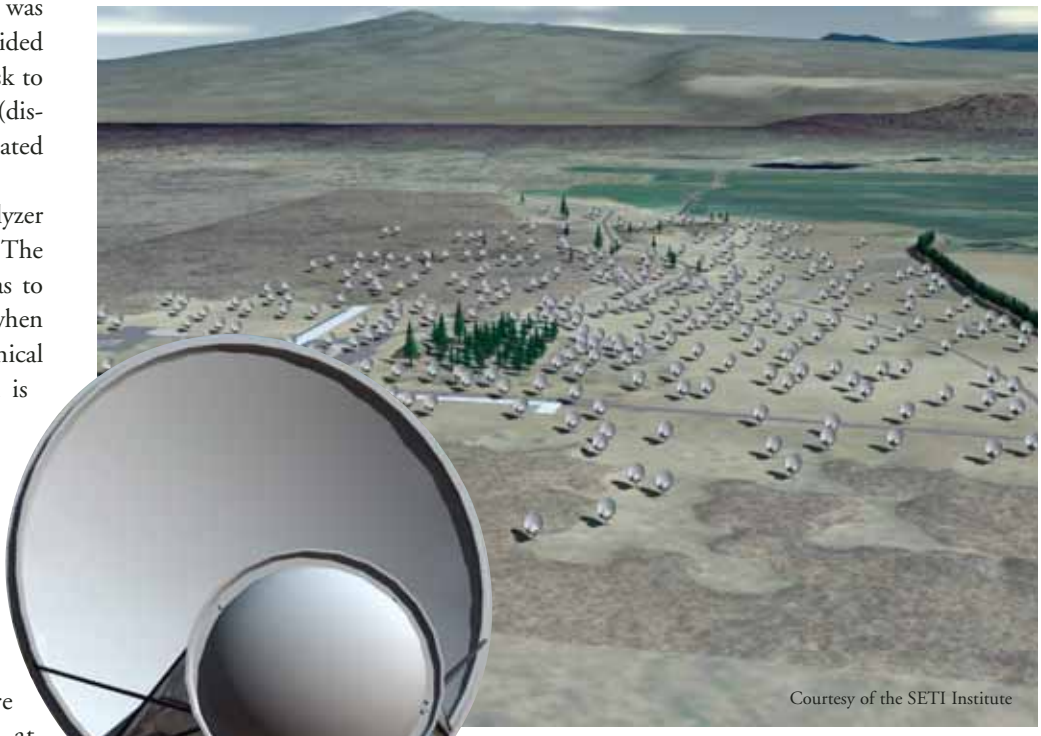


Figure 5 – Artist's conception of Allen Telescope Array with offset Gregorian dish antennas

Employing the same piggyback strategy used at Arecibo, SETI researchers will scan for ET wherever other radio astronomy research projects aim the array.

Although these dishes can be "stamped out like hot tubs" very cheaply, Werthimer says, no one has ever attempted to build a giant telescope from so many small dishes before. The costs of the electronics and signal processing technology required to manage such an array were prohibitively high. The signal processing computation grows as the square of the number of antennas, he explains. But, "Thanks to Virtex-II Pro™ and Spartan™-3 chips, the peta-op per second signal processing costs will comprise only a tiny fraction of the total system costs."

Each telescope (Figure 6) will have an extremely wide-band dual polarization feed, covering 0.5 GHz to 11.2 GHz. This feed will drive a pair of wide-band, low-noise amplifiers that output their signals to a pair of analog optical fiber laser modulators. All telescope fibers will be routed to the central electronic lab where the SETI signal processors and image processors will be located.

To make maps of the radio sky, the FPGA-based "imager" for the antenna array must process real-time data at the rate of one terabit per second (10^{12} bits/sec) and compute one peta-op per second (10^{15} ops/sec) – that's 20 times faster than the SETI@home supercomputer, Werthimer declares.

First, each telescope signal will be digitized and broken up into 1,024 spectral components by means of a polyphase filter bank. The resulting data from each telescope will then be sent to a "corner turner" that will re-order this data by frequency channel.

Next, each telescope "pair" will be cross-correlated and integrated (there are N*(N-1)/2 "pairs" of telescope signals to correlate). Then the data will be "2D Fourier transformed" to produce an image. All signal processing and data routing will be implemented using several thousand Virtex II-Pro platform FPGAs in Rack 2 and Spartan-3 FPGAs in Rack 4, Werthimer says.

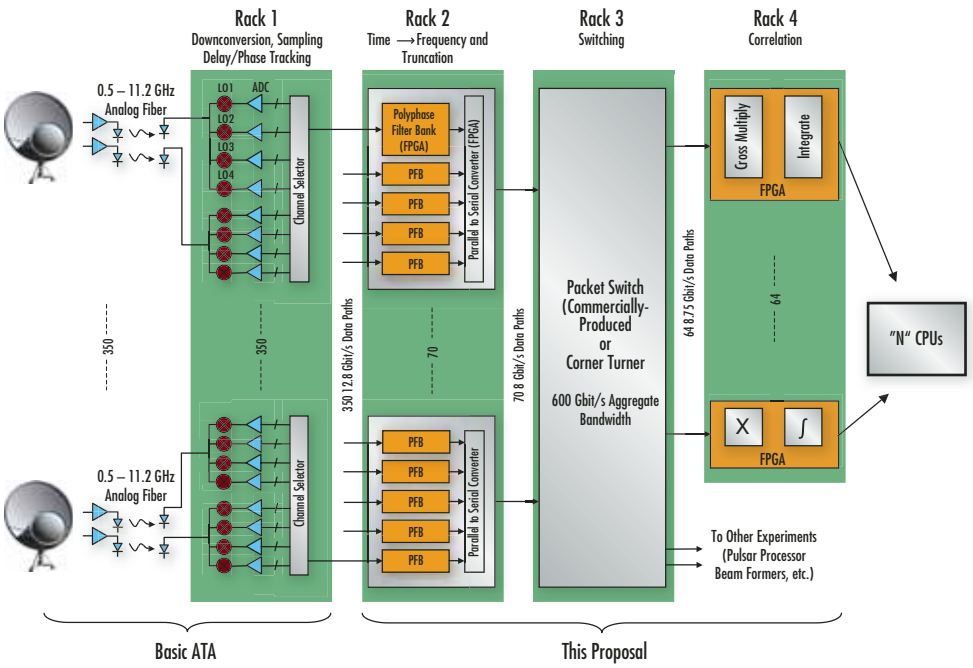


Figure 6 – Proposed Allen Telescope Array image former

Turning Science Fiction into Fact

SETI scientists operate on the assumption that there are other intelligent civilizations "out there." Otherwise, why look? So, the question is not "if," but "how many?" And where? And when will we find them?

Xilinx University Program

Xilinx donations to SETI research date back to 1988. Patrick Lysaght, senior director of Xilinx Research Labs, which includes the Xilinx University Program (XUP), reports that within the past five years, Xilinx has donated more than $1.5 million in hardware, software, and technical support to UC Berkeley's SETI research.

"The search for extraterrestrial intelligence is a tremendously exciting adventure at the frontiers of science," Lysaght says. "Xilinx FPGAs are uniquely suited to the huge computing demands of projects such as SETI."

Werthimer is unrestrained in his gratitude to XUP: "The Xilinx University Program has really made this whole thing possible. Not only did you guys develop the core technology that we needed to process 168 million channels simultaneously, but you have this generous program that gives us the software and the chips to build SERENDIP V."

Werthimer's gratitude goes beyond words. In a memo to XUP last year, he wrote: "We'd like to reciprocate as best we can for all the wonderful stuff you've donated." For example: "We have a flow-through dual FFT and polyphase filter design that Xilinx and its customers are welcome to use. It uses one-fourth of the memory that the Xilinx FFT IP core uses, and it calculates two complex FFTs simultaneously at 240 million samples per second."

> "The Xilinx University Program has really made this whole thing possible."
>
> – Dan Werthimer, director, SERENDIP SETI Program

Contact

Why are companies like Xilinx, Sun Microsystems, Intel™, Toshiba™, Hewlett Packard™, Quantum™, Network Appliance™, Fujifilm™ – as well as non-profit organizations like the SETI Institute, the SETI League, and The Planetary Society – contributing millions of dollars worth of technology and expertise to support the search for ET? Xilinx CEO Wim Roelandts puts it this way: "You can say, well, maybe it is science fiction, but to prove it isn't, you need state-of-the-art technology. This is probably the ultimate science problem we must solve. The intellectual challenge is enormous. And that is what is exciting."

Within the SETI community, the word "excited" has become almost a code word for the discovery of ET. Pretty much everybody uses the word "excited" when they describe how they'll feel when a signal from ET is scientifically confirmed. That means replicated results from other, independent observatories, such as those in Australia, Italy, France, and Argentina.

"We actually want to look at the sky many, many times," Werthimer explains. "One of our most robust algorithms is: Did you see the signal again, in same place, when the telescope comes back to the same place in the sky? Do we see it in the same place at the same frequency? – that's what gets us really excited."

Unfortunately, in the last 44 years of serious, scientific research, nobody's found anything to get all that excited about.

"I'm optimistic," Werthimer says. "I think we might find ET in our lifetimes, but I think, right now, we'd be very lucky to find ET. So, I'm sort of counting on Moore's Law. If Moore's Law keeps going, and if Xilinx keeps on making faster and better chips, the better the chances we have of finding an extraterrestrial radio signal."

Two Scenarios

"There are sort of two scenarios for contact with ET," Werthimer reasons. "One is that we find a signal, and we aren't really able to decode it. It could just be a navigational beacon – there's no information. All we would know is that they're out there.

> "I'm optimistic. I think we might find ET in our lifetimes."
>
> – Dan Werthimer

"The more scary scenario to me," Werthimer continues, "is that we might receive a direct broadcast with a huge amount of information content in the ET signal – and that could be used in good ways or bad ways."

To prevent the potential abuse of extraterrestrial intelligence, virtually all SETI research organizations – including the SETI SERENDIP Program, the SETI Institute, the SETI League, and The Planetary Society, to name just a few – have endorsed Article XI of the United Nations Treaty on Principles Governing the Activities of States in the Exploration and Use of Outer Space. In part, Article XI decrees that the discoverers of ET must "... inform the Secretary General of the United Nations, as well as the public and the international scientific community, to the greatest extent feasible and practicable, of the nature, conduct, locations, and results ..." of the discovery.

Furthermore, the treaty calls for a transnational decision on whether to reply to ET – and if so, what to say.

Conclusion

We may discover ET any time from now to never. And it's impossible to predict exactly how humanity will react to scientifically validated proof that life as we know it – isn't.

For many years, the late astronomer-philosopher Carl Sagan was the standard-bearer for SETI research. He gave as much thought to the implications of finding ET as he did to the technology of the search for ET. In a 1978 essay, he wrote: "The search for extraterrestrial intelligence is the search for a generally acceptable cosmic context for the human species. … It is difficult to think of another enterprise within our capability, and at relatively modest cost, which holds as much promise for the future of humanity." Σ

Hyperlinks to SETI Research

Allen Telescope Array:
www.seti.org/science/ata.html

Arecibo Observatory:
www.naic.edu/bigtable.htm

Origins: Astrobiology: The Search for Life:
www.exploratorium.edu/origins/arecibo/

SETI@home:
setiathome.ssl.berkeley.edu

SETI at the University of California, Berkeley:
seti.berkeley.edu

SETI Institute:
www.seti.org

SETI League:
www.setileague.org

The Planetary Society:
seti.planetary.org

"The Quest for Extraterrestrial Intelligence" by Carl Sagan:
www.bigear.org/vol1no2/sagan.htm

Xilinx University Program:
www.xilinx.com/univ/

Protocols for Contact with ET

Treaty on Principles Governing the Activities of States in the Exploration and Use of Outer Space:
www.oosa.unvienna.org/SpaceLaw/outersptxt.htm

SETI Institute:
www.seti.org/science/principles.html

SETI League:
www.setileague.org/general/protocol.htm

The Planetary Society:
seti.planetary.org/Contact/AfterTheDetection.html

Celebrating 20 Years of Innovation

As Xilinx marks its 20th anniversary, a Xilinx Fellow recalls the birth of programmable technology.

Xilinx Fellow, Bill Carter, in the Xilinx Hall of Patents.

by Xilinx Staff

Twenty years in any industry is a long time; in the lightning-paced semiconductor business, it can seem like a lifetime. But for Bill Carter, the first chip designer hired by Xilinx shortly after the company was founded, programmable technology really is just entering its adolescence.

"I'm surprised at how far we've come in a relatively short period of time," admits the understated Xilinx Fellow, who doubles as the unofficial company historian (which essentially means historian for an entire industry). "But we've got a long way to go to reach maturity, simply because the application potential for programmable technology is so vast and still largely untapped."

Indeed, compared to its more staid silicon cousins such as microprocessors and memory – the embodiments of fixed architectures – programmable technology is still a wild-haired teenager, in some ways battling for respect and searching to find itself. But no one can deny the impact FPGAs or programmable logic devices (PLDs) – today a multi-billion-dollar market – have had on the semiconductor industry and on products that touch our lives every day.

In the context of an ever-changing electronics industry and relentless improvements in semiconductor technology, the unique benefits of programmability seem destined to be a cornerstone of innovation and progress for years to come.

Challenging a Mindset

Of course, no one could have predicted that 20 years ago. When the founding fathers of Xilinx – Bernie Vonderschmitt, Ross Freeman, and Jim Barnett – launched their oddly named start-up venture (perhaps appropriately in the Orwellian-prophesized year 1984), the semiconductor world was a vastly different place than it is today. The PC, destined to be the heavyweight champion of silicon consumption, was just emerging from Silicon Valley labs into commercial viability. The Internet was an arcane communication link for scientists and the government, wireless telephones were about the dimension of a cinder block, and Bill Gates still had to work for a living.

More importantly to the Xilinx founders, many of the engrained ways of thinking and doing business in the semiconductor industry, while seemingly permanently rooted, were in their minds becoming a bit misguided and shortsighted.

"It was Ross Freeman, really, who had the radical notion that transistors are free," remembers Carter. "In those days, gates were precious and everyone thought, 'fewer transistors is better.' Ross challenged all that and saw the potential for leveraging the available real estate on chips to allow customers to customize their devices. It was contrary to everything most chip designers had learned, including me."

Of course, Moore's Law saw to it that eventually semiconductor designers would have more transistors than they knew what to do with.

The other tenet of the semiconductor industry that Xilinx immediately challenged was the concept of a company owning its own manufacturing capability. Fabs were then, as they are today, an expensive proposition, but also considered a closely guarded competitive advantage for chip companies.

Vonderschmitt, through past relationships and a straightforward and fair business style, managed to convince Japan's Seiko Corp. that allowing Xilinx-designed chips to be built in their fabs was a good idea for both companies. Little did he know that this would launch a whole new approach that today is commonplace – the fabless semiconductor company.

"That part of our strategy was borne out of practicality more than anything else. We knew we didn't have enough money to build a fab, and we certainly didn't have enough customers to fill one," says Carter. "Bernie was able to put together a deal that was truly win-win for both sides."

Perhaps the only thing that would be familiar to today's semiconductor participants was that in 1984, the industry was in a slump. Undeterred, the Xilinx founders shopped around an ambitious business plan, ultimately securing just over $4 million in funding to launch their venture. Their initial plan called for first silicon by mid-1985, and $200 million in sales by 1990 (a figure they would ultimately reach in 1993). They quickly assembled a team of software and chip design experts that shared their vision (some, such as Carter, took no small amount of convincing) and set out to change the world, transistor by transistor.

> "Programmable devices were not a new concept in 1984, but they were anything from mainstream."

It's the Interconnect!

Programmable devices were not a new concept in 1984, but they were anything from mainstream. Programmable logic arrays (PLAs) had been around since the 1970s, but were considered quirky, slow, and hard to use. In the early 1980s, configurable programmable array logics (PALs) had begun to emerge, offering a limited ability to implement flip-flops and look-up tables enabled by crude software tools.

Manufacturing processes were in the 2-3 micron range, and transistors still were the key to performance. PALs were seen as a replacement to small-scale integration/medium-scale integration (SSI/MSI) glue-logic parts, and slowly gained favor with the more aggressive engineering set.

But programmability remained a foreign and risky proposition for most, further compounded by attempts in the mid-1980s at "mega PALs," which suffered from critical drawbacks in terms of power consumption and process scalability that would ultimately limit broader adoption.

Xilinx's technical strategy was based around Freeman's belief that for many applications, flexibility and customization would be an attractive feature if implemented correctly – perhaps only for prototyping at first, but potentially also as a replacement to more rigidly-defined custom chips. ASIC design was starting to take root, pushed along by better design tools such as simulation and other computer-aided engineering (CAE) capabilities, as well as the increasing spectrum of applications for which silicon technology could be used. But Xilinx saw an opportunity to offer an even more customizable approach.

The linchpin of their innovation was the idea of programmable interconnect. In fact, Xilinx's name is drawn from this concept: The Xs at each end of Xilinx represent programmable logic blocks (or configurable logic blocks [CLBs]). The -linx represents programmable interconnect connecting the logic blocks together.

The founders took a page from printed circuit board (PCB) design (and a precursor to today's system-on-chip [SoC] design) and envisioned arrays of custom logic blocks surrounded by a perimeter of I/Os, all of which could be assembled arbitrarily, thus overcoming the scalability issue PALs had run into (which were constrained by fixed I/Os).

The concept borrowed from the increasingly popular gate array technique, but supported the notion of post-manufacturing customization. Programming would be enabled by a set of graphical and intuitive PC-based design tools, and customers could quickly change the functionality of the chips.

Best of all, it was scalable to new manufacturing processes, a benefit perhaps even the founders may not have fully appreciated

"...the threshold for where FPGAs make sense is getting higher and higher."

when the first chip rolled off the production lines in 1985, containing 85,000 transistors in a 2-micron process. The XC2064 was conservatively designed even for that era, containing 64 logic blocks and probably no more than 1,000 gates. However, Xilinx pioneers were determined to "only take risks with the concept, not the technology." And the rest, as they say, is history.

Fast and Furious Progress

By the third generation of Xilinx FPGAs, the 4000 series, people were beginning to take programmable technology seriously. The XC4003 contained 440,000 transistors and was implemented in a much more leading-edge 0.7-micron process.

The performance and capacity of FPGAs were closing in on fixed architecture alternatives. In fact, FPGAs were beginning to be looked at as good vehicles for process development by manufacturers (at this point, dedicated foundries had emerged as a viable component of the semiconductor supply chain).

"By the mid 90s, we were secure enough in the concept that we could become more aggressive with how it was implemented," Carter explains. "Plus, it turns out that FPGAs provided excellent observability into new processes, so they were being used in a way that memories had been to qualify each new generation. That put us on the leading edge, to the point now that I believe we have surpassed Moore's Law."

Xilinx had shipped its one-millionth device by 1989; a public offering in 1990 established the company's sustainability. More success came quickly after that as it rode the wave of silicon proliferation. By 1995, the company was ranked as the 10th largest ASIC supplier, revenues were close to a half billion dollars, and the company had grown to more than 1,000 employees, with offices around the world.

A steady stream of innovation and increasingly competitive capabilities won Xilinx new customers across a variety of application spaces – and many new converts to the "programmable way." The Xilinx product line expanded into high-end, high-volume, and low-power variations, giving customers even more freedom of choice. Its Spartan™ family set a new price/performance standard in 1998 when it was fist introduced, while the high-end Virtex™ device became the first million-gate FPGA in that same year, enabled by aggressive use of 0.25-micron processes from Xilinx's manufacturing partners.

By 2000, sales had topped $1 billion and Xilinx was among the first semiconductor companies to have a reachable roadmap to 90 nm processes and 300 mm wafer manufacturing.

Heading into its 20th year, Xilinx continues to set new standards, this year releasing the world's first one-billion transistor device – a platform FPGA that not only breaks down another barrier in terms of complexity and integration, but establishes an innovative approach to bring the power and flexibility of FPGAs to a variety of applications.

The Value is in the Words

Today Xilinx ranks as the fourth largest ASIC supplier, demonstrating that programmability is not only here to stay, but is quickly becoming the customizable alternative of choice among product developers.

"ASICs are already dead as far as I'm concerned," declares Carter. "The technical barriers are gone. FPGAs are fast enough, big enough, and economically viable in high-volume applications. It's really just a perception issue standing in the way of more widespread adoption."

Carter feels the mindset challenges are slowly fading away as older engineers are replaced by a new generation of designers. Plus, FPGAs offer undeniable benefits in a world where products go out of style in months, an alphabet soup of standards changes at a dizzying pace, and companies require multiple variations of the same core design.

"Nothing can exploit the improvements in process technology; nothing can leverage the available transistors, like FPGAs can. The truth is that the value-add in laying out transistors, a traditional advantage for semiconductor companies, is diminishing. The way of the future will be to think about delivering intellectual property to more people, not in optimizing layouts," Carter says.

Using a publishing analogy, Carter explains, "The value is in the words, not the ink and paper. We can supply the foundation on which to build an infinite combination of words that can be customized for any number of applications."

When asked if there are any applications that can't benefit from using programmable technology, Carter replies, "I really can't think of any. Put it this way – the threshold for where FPGAs make sense is getting higher and higher."

In fact, Carter and his colleagues at Xilinx envision FPGAs as the foundation to a whole new approach to product development, analogous to current software development approaches. Designers would use FPGAs to do weekly design builds, running tests on both hardware and software, essentially having access to a dynamic prototype. In some ways this returns FPGAs to their original roots, but the difference is that now you can "ship the prototype," says Carter.

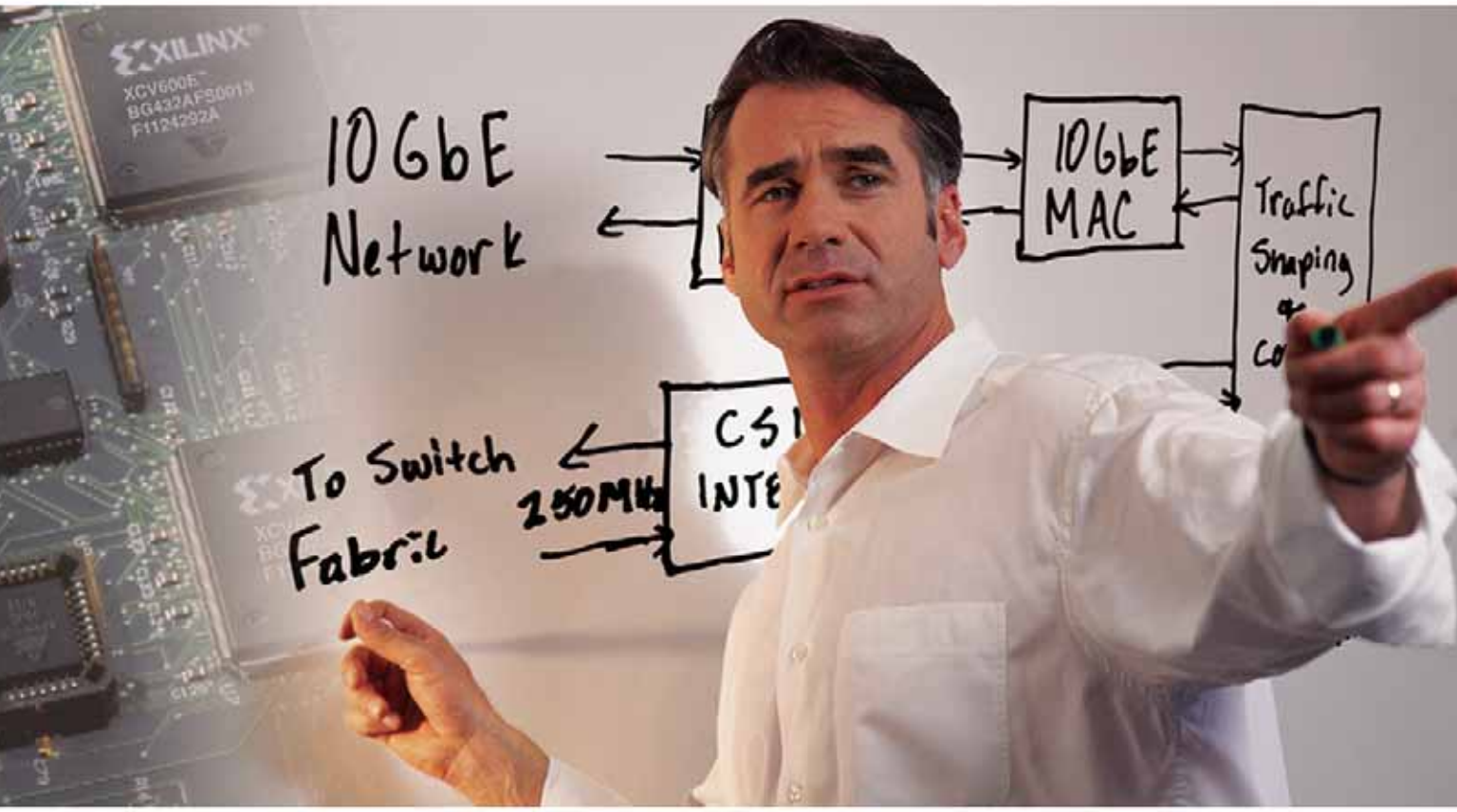
Beyond Silicon

The full potential of programmable technology may be beyond any single person's wildest imagination. Exciting and unthought-of breakthroughs will come when research disciplines are crossed, such that ideas from biology or genetics, for example, are merged with silicon physics to create bold new applications.

Programmable biological systems? Not as crazy as you may think for an uninhibited adolescent who is just coming into its own. Σ

Xilinx Education Services.

Expertise across the board.



www.xilinx.com/education

Xilinx Education Services provide the expertise you need to get your designs completed on time, on budget, and with optimal performance.

- Reduced time to knowledge
- Improved design efficiency
- Lower overall development costs

Public, private, and online classes are available worldwide, so you can find the one that best fits your specific language, location, and application needs.

Finish Faster.

Xilinx delivers customer service and support through every step of the design process—around the clock and around the world—to help you get to market faster. From technical support, consultative services, and dedicated design assistance to online support and training, you can find it all at www.xilinx.com/services.

For more information, visit **www.xilinx.com/education**

XILINX®

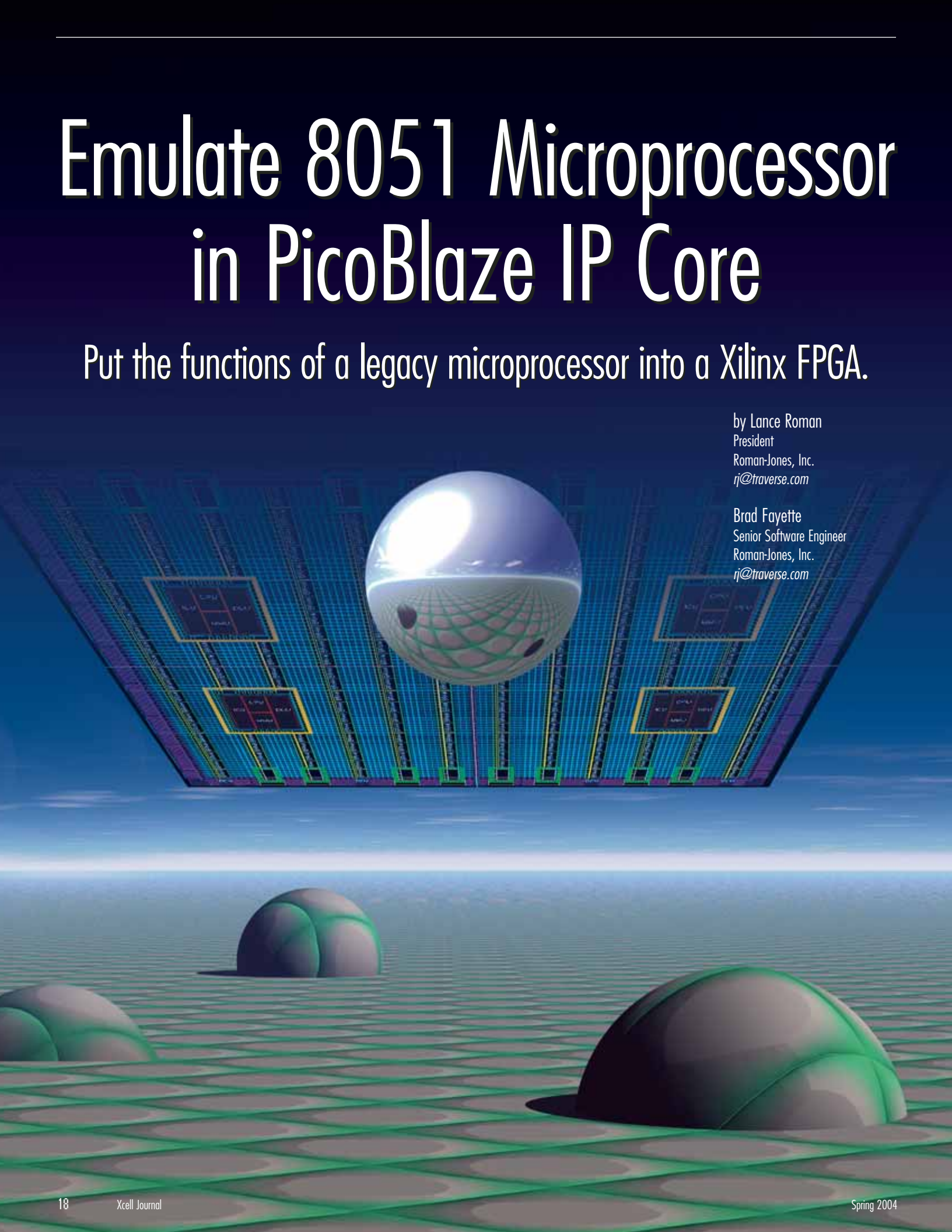
The Programmable Logic Company℠

Emulate 8051 Microprocessor in PicoBlaze IP Core

Put the functions of a legacy microprocessor into a Xilinx FPGA.

by Lance Roman
President
Roman-Jones, Inc.
rj@traverse.com

Brad Fayette
Senior Software Engineer
Roman-Jones, Inc.
rj@traverse.com



How do you put a one-dollar Intel™ 8051 microprocessor into an FPGA without using 10 dollars' worth of FPGA fabric?

The answer is emulation. Using software emulation, Roman-Jones Inc. has developed a new type of 8051 processor core built on a Xilinx 8-bit, soft-core PicoBlaze™ (PB) processor. This "new" PB8051 is more than 70% smaller than competing soft-core implementations – without sacrificing any of the performance of this legacy part. The PB8051 is a Xilinx AllianceCORE™ microprocessor built through emulation.

The Legacy of the 8051

The Intel 8051 family of microprocessors – probably one of the most popular architectures around – is still the core of many embedded applications. This processor just refuses to retire. Many designers are using legacy code from previous projects, while others are actually writing new code.

The 8051 architecture was designed for ASIC fabric. It is not efficient in an FPGA, resulting in excess logic usage with marginal performance.

FPGA microprocessor integration is a solution for older 8051 products undergoing redesign to eliminate obsolesce, lower costs, decrease component count, and increase overall performance.

The new FPGA-embedded PB8051 designs allow you to take advantage of existing in-house software tools and your own architecture familiarity to quickly implement a finished design. The integrated PB8051 can be customized on the FPGA to exact requirements.

Processor Emulation

Programmers have used microprocessor emulation for many years as a software development vehicle. It allows programmers to write and test code on a development platform before testing on target hardware.

This same concept can be practical when the target microprocessor architecture does not lend itself to efficient implementation and use of FPGA resources.

The features of our PB8051 emulated processor include:

- Smaller Size – Traditional 8051 implementations range from 1,100 to 1,600 slices of FPGA logic. The PicoBlaze 8051 processor requires just 76 slices. Emulation hardware requires 77 slices. Add another 158 slices for two timers and a four-mode serial port, and you have a total of a 311 slices. This is a reduction of more than two-thirds of the FPGA fabric of competing products.
- Faster – At 1.3 million instructions per second (MIPS), the PB8051 is faster than a legacy 8051 (1 MIPS) running at 12 MHz. Compare this to 5 MIPS with a 40 MHz Dallas version or 8 MIPS with a traditional FPGA implementation – which takes up more than three times as much FPGA fabric as the PB8051. The PicoBlaze processor itself runs at a remarkable 40 MIPS using an 80 MHz clock.
- Software Friendly – You can write C code or assembler code with your present software development tools to generate programs. You can also run legacy objects out of a 27C512 EPROM.
- Easy Hardware – You can use VHDL or Verilog™ hardware description languages to instantiate the 8051-type core. Hook up block RAM or off-chip program ROM, and you're ready to go.
- Low Cost – The PB8051 is $495 with an easy Xilinx SignOnce IP license.

> This "new" PB8051 is more than 70% smaller than competing soft-core implementations – without sacrificing any of the performance of this legacy part.

Architecture of Emulated 8051

As shown in Figure 1, the architecture of an emulated processor has several elements. Each element is designed independently, but together, they act as a whole.

PicoBlaze Platform

The PicoBlaze host processor is the heart of the emulated system and defines the architecture of the PB8051 emulated processor core. It comprises:

- PicoBlaze Code ROM – Block ROM contains the software code to emulate 8051 instructions.
- Internal Address/Bus – The PicoBlaze peripheral bus is a 256-byte address space accessed by PicoBlaze I/O instructions. It allows the PicoBlaze processor to interface with RAM, emulation peripherals, timers, and the serial port. This bus is internal to the core and thus hidden from the user.
- Emulation Peripherals – Not all emulation tasks can be done in software and still meet the performance requirements for your particular application. Specialized hardware assists the PicoBlaze code to perform tasks that are time-consuming, on a critical execution path, or both.

A good example is the parity bit in the PSW (program status word) register that reflects 8051 accumulator parity. This function must be performed for every 8051 instruction executed, and would take several PicoBlaze instruction cycles to perform. It is, therefore, done in hardware.

- Instruction Decode ROM – This is a key emulation peripheral that is used to decode 8051 instructions to set PicoBlaze routine locations and emulation parameters.

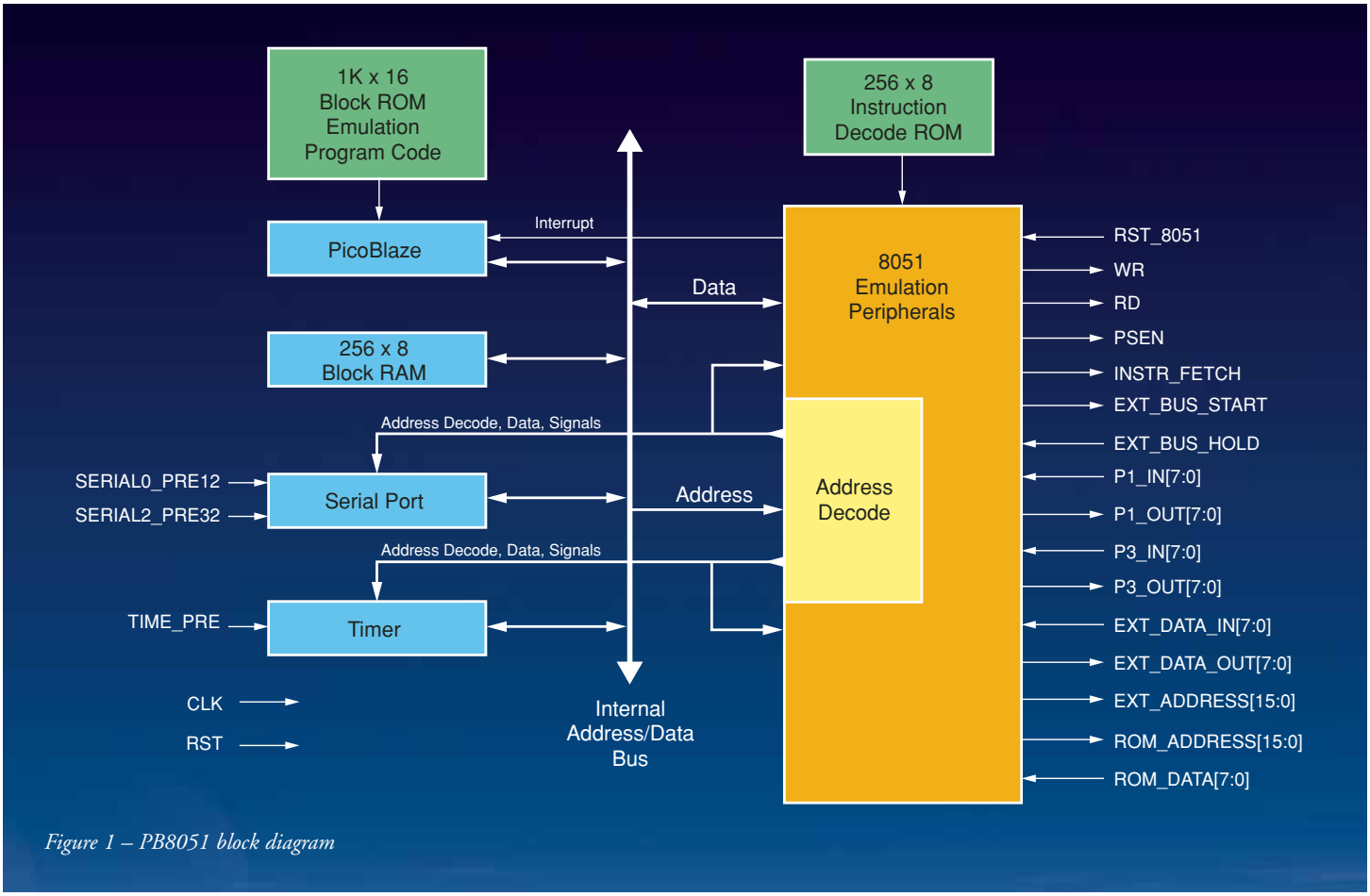


Figure 1 – PB8051 block diagram

- Address Decode – A significant amount of the emulation peripheral hardware is dedicated to simple address decode of the PicoBlaze peripheral bus. This address space is for the PicoBlaze processor only and is insulated from the 8051 application.
- Block RAM – 256 bytes are available to 8051 internal RAM and some 8051 registers. You can access this block RAM via 8051 instructions.
- Serial Port and Timer – The actual 8051 timer and multi-mode serial port were best done in hardware instead of trying to implement these functions in software. Clock prescaling inputs are provided so that these functions can run at a clock rate independent of the emulated system.

PicoBlaze Emulation Software

A 1K x 16 block ROM holds the PicoBlaze code that performs the actual emulation. The emulation program is carefully constructed in very tight PicoBlaze assembler code, optimized for speed and efficiency.

The emulation program is divided into several segments:

- Instruction Fetch – An 8051 bus cycle is simulated to fetch the next instruction from 8051 program memory (64 Kb size), which may be on-chip block RAM or off-chip EPROM (such as the 27C256).
- Instruction Decode – Fetched instructions are decoded to determine the addressing mode and operation.
- Fetch Operands – Depending upon the addressing mode, additional operands are fetched for the instruction from program memory, internal RAM, or external RAM.
- Instruction Execution – An instruction performs the desired operation and updates emulated register contents, including affected 8051 PSW flags.
- Scan Interrupts – This function determines if an interrupt is pending, and if so, services it. An interrupt window is generated at the end of every emulated instruction when required.

In addition to PicoBlaze code, Java™ software utilities process symbols taken from PicoBlaze listings (.LOG files) into a ROM table. These .LOG files are used to decode 8051 opcodes.

This program also produces the .COE files used by the Xilinx CORE Generator™ system to create PicoBlaze code ROM. All of this is transparent to designers integrating with the PB8051.

User Back-End Interface

What gives the PB8051 its "hardware flavor" is the user back-end interface, where you interface your logic design with the emulated 8051 processor. The back-end interface is part of the emulation peripherals, controlled by the PicoBlaze platform.

What gives the PB8051 its "hardware flavor" is the user back-end interface, where you interface your logic design with the emulated 8051 processor.

Just as the 8051 processor family has many derivatives to define port, functionality, features, and pinout, the back-end interface serves the same function. The PB8051 has a back-end interface that resembles the generic 8031, the ROM-less version of the 8051.

Roman-Jones Inc. customizes back-end interfaces to meet your exact 8051 needs, such as removing an unused serial port or adding an I^2C port to emulate the 80C652 derivative.

Designing with the PB8051

Incorporating the PB8051 into the rest of your design is easy, because it comes with reference designs and examples of Xilinx integrated software design (ISE) projects.

Hardware Considerations

Figure 1 illustrates the signal names available on the user back-end interface. A VHDL or Verilog template provides the exact signal names – many of which are already familiar to 8051 designers. A few new types of signals exist, including:

- Pre-scales used by the timer and serial port to set counting and baud rates.
- One-clock-wide read/write strobes to read and write external memory space. There is also a program store enable (PSEN) strobe for the 8051 code memory.
- Bus start and hold signals used to insert wait states for external memory cycles that cannot be completed in one system clock cycle.

The PB8051 is instantiated as a component into your top-level design. You determine if the 8051 program resides in on-chip block RAM or off-chip EPROM. Peripherals can be hooked up to either the P1/P3 port lines or to the external address and data buses. For convenience, the address and data lines for program and data memory spaces are separated, so conventional multiplexer circuitry is not needed.

If your entire design, including the 8051 program, resides on the FPGA, simply set the EXT_BUS_HOLD to "low" to take full advantage of running at clock speed. If you elect to use an off-chip EPROM or have slow peripherals, wait states can be inserted by asserting EXT_BUS_HOLD at "high." One of our reference designs illustrates wait state generation.

Xilinx Implementation Considerations

There are few implementation considerations other than the need to place the PB8051 design netlist into your project directory and instantiate it as a component in your VHDL or Verilog design; ISE software will do the rest. ISE schematic capture is also supported.

Simulation Considerations

We've included a behavior simulation model of the PB8051 for adding (along with the rest of your design) to your favorite simulator. Modeltech and Aldec™ simulators have been tested for correct operation. Post place-and-route or timing simulations follow a conventional design flow.

You will enjoy watching your 8051 instruction execution flow go by on the simulation waveforms. This makes behavioral debugging straightforward, fast, and easy. We've provided a reference test bench with example waveform files.

8051 Software Considerations

To generate your 8051 programs the way you always have, use your favorite C compiler or assembler and linker (if necessary) to produce the same Intel hex format file that you would use to burn an EPROM. You can even use an existing hex file, because the PB8051 looks like a regular 8051 as far as software code is concerned. An Intel hex to .COE utility is included for those designs that put the 8051 software into on-chip block RAM. On-chip program storage provides maximum speed performance.

Test and Debug Considerations

We recommend you use design tools to quickly and easily test and debug your design. The most useful tool will be an HDL simulator, such as Modeltech or Aldec programs. Most problems and bugs can be solved at the behavioral level. For interactive debugging, the Xilinx ChipScope™ integrated logic analyzer has proved to be the tool of choice. At the current time, no source code debugger tools are available for the PB8051.

Designer's Learning Curve

Designers should have some experience in 8051 hardware/software and FPGA design before attempting to consolidate the two. The PB8051 core is designed for ease of use and integration.

Your 8051 hardware and software expertise should include hardware understanding of the part and experience in writing 8051 code using software development tools. The basic design flow using the PB8051 is identical to the normal packaged processor flow.

Integrating the PB8051 onto the Xilinx part is much the same as instantiating a core using the Xilinx CORE Generator tool. If you are familiar with VHDL or Verilog language, and have a couple of Xilinx designs under your belt, you're good to go.

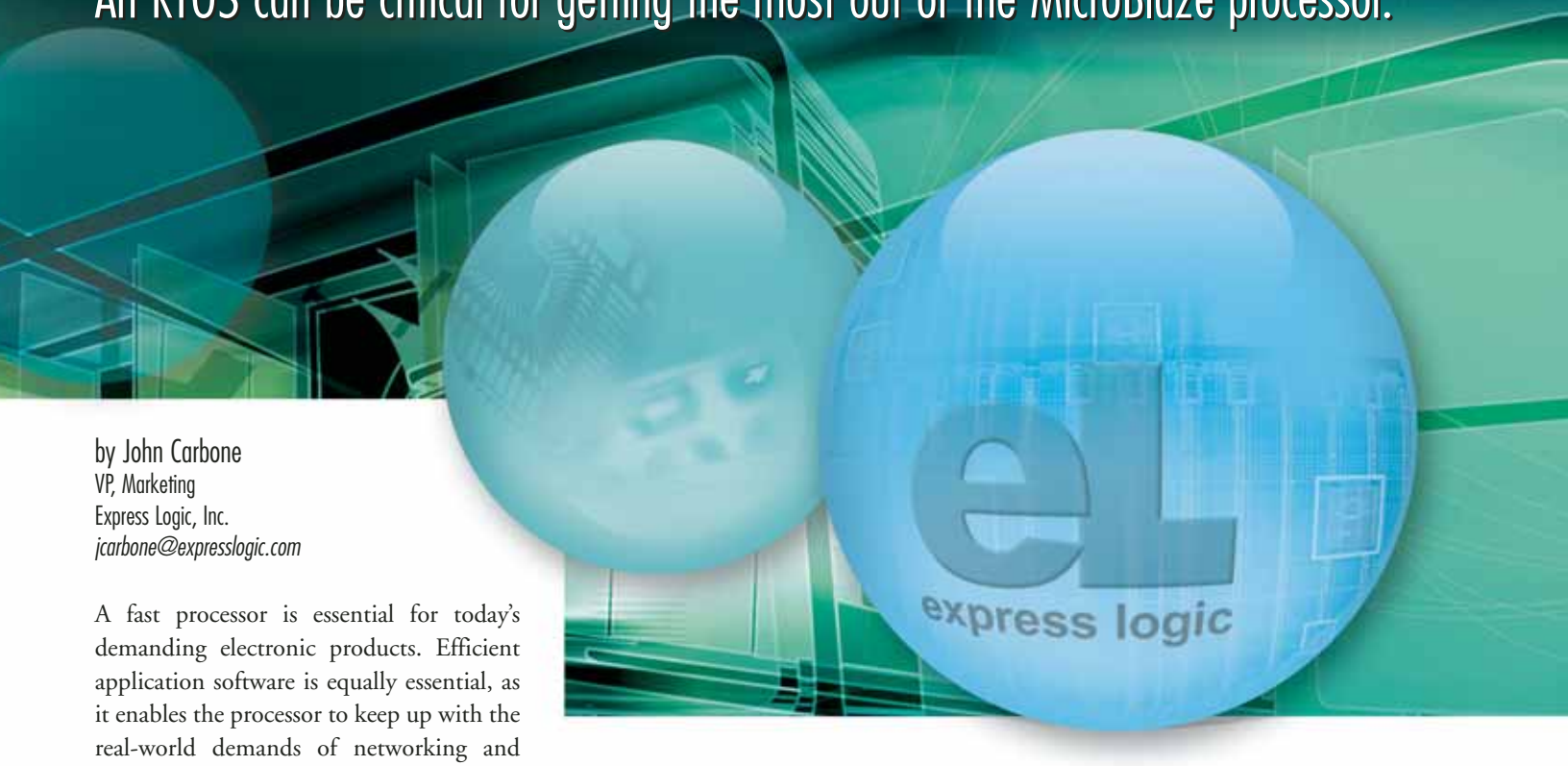
Conclusion

Integrating microprocessors onto FPGAs through emulation, as illustrated with the PB8051, is a viable alternative to a full hardware functional design. The advantage of FPGA fabric savings correlates to reduced parts cost.

Integrating the PB8051 processor into your design yields lower component count, easier board debugging, less noise, and optimum performance of the peripheral/8051 micro-interface, because both are in an FPGA. For more information about the PB8051 microcontroller, visit *www.roman-jones.com/rj2/PB8051Microcontroller.htm.* Σ

Optimize MicroBlaze Processors for Consumer Electronics Products

An RTOS can be critical for getting the most out of the MicroBlaze processor.



by John Carbone
VP, Marketing
Express Logic, Inc.
jcarbone@expresslogic.com

A fast processor is essential for today's demanding electronic products. Efficient application software is equally essential, as it enables the processor to keep up with the real-world demands of networking and consumer products.

But what about the real-time operating system (RTOS) that handles your system interrupts and schedules your application software's multiple threads?

If it's not optimized for the processor you're using, and isn't efficient in its handling of external events, you'll end up with only half a processor to do useful work.

An efficient RTOS should do more than just handle multiple threads and service interrupts. It must also have a small footprint and be able to deliver the full power of the processor used by your application software. Express Logic's ThreadX® real-time operating system is just such an RTOS.

The ThreadX RTOS has now been optimized to support the Xilinx MicroBlaze™ soft processor, and is available off-the-shelf for your next development project.

A Good Fit

Let's look at some of the reasons why the ThreadX RTOS is a good fit with the MicroBlaze processor.

Size

The ThreadX RTOS is only about 6 KB - 12 KB in total size for a complete multitasking system. This includes basic services, queues, event flags, semaphores, and memory allocation services.

Its small size means that more of your memory resources will be preserved for use by the application and not absorbed by the RTOS. This will result in smaller memory requirements, lower costs, and lower power consumption. Table 1 shows code sizes for various optional components of the ThreadX RTOS on the MicroBlaze processor.

Efficiency

The ThreadX RTOS can perform a full context switch between active threads in a mere 1.2 µs on a 100 MHz Xilinx Virtex-II Pro™ FPGA. This is particularly critical in applications with high interrupt incidences, such as network packet processing.

The ability of the ThreadX RTOS to handle interrupts efficiently means that your system can handle higher rates of external events such as TCP/IP packet arrivals, delivering greater throughput.

Speed

The ThreadX RTOS responds to interrupts and schedules required processing in less than 1 microsecond. The single-level interrupt processing architecture of the ThreadX RTOS eliminates excess overhead found in many other RTOSs.

This keeps the ThreadX RTOS from gobbling up valuable processor cycles with housekeeping tasks, thus leaving more processor bandwidth for your application. Thus, you can configure a less expensive processor or add more features to your application without increasing processor speed and cost.

Table 2 shows execution times for various ThreadX services measured according to the following definitions:

- **Immediate Response** (IR): Time required to process the request immediately, with no thread suspension or thread resumption.
- **Thread Suspend** (TS): Time required to process the request when the calling thread is suspended because the resource is unavailable.
- **Thread Resumed** (TR): Time required to process the request when a previously suspended thread (of the same or lower priority) is resumed because of the request.
- **Thread Resumed and Context Switched** (TRCS): Time required to process the request when a previously suspended thread with a higher priority is resumed because of the request. As the resumed thread has a higher priority, a context switch to the resumed thread is also performed from within the request.
- **Context Switch** (CS): Time required to save the current thread's context, find the highest priority ready thread, and restore its context.
- **Interrupt Latency Range** (ILR): Time duration of disabled interrupts.

Ease of Use

All services are available as function calls, and full source code is provided so you can see exactly what the ThreadX RTOS is doing at any point.

Many other commercial RTOS products are massive collections of unintelligible system calls with non-intuitive names. The ThreadX RTOS uses service call names that are orderly, simple in structure, and ultimately easier to use. This enables shorter development times and faster time to market.

Fast Interrupt Handling

The ThreadX RTOS is optimized for fast interrupt handling, saving only scratch registers at the beginning of an interrupt service routine (ISR), enabling use of all services from the ISR, and using the system stack in the ISR.

ThreadX Instruction Area Size (in bytes)	
Core Services (required)	4,804
Queue Services	3,200
Event Flag Services	1,720
Semaphore Services	932
Mutex Services	2,540
Block Memory Services	1,144
Byte Memory Services	1,876

Table 1 – Code size of various ThreadX services

ThreadX Service Call Execution Times				
Service	**IR**	**TS**	**TR**	**TRCS**
tx_thread_suspend	1.0 µs	2.0 µs		
tx_thread_resume			1.0 µs	2.8 µs
tx_thread_relinquish	0.5 µs			1.9 µs
tx_queue_send	0.9 µs	2.4 µs	1.6 µs	3.4 µs
tx_queue_receive	0.9 µs	2.3 µs	2.5 µs	4.2 µs
tx_semaphore_get	0.3 µs	2.3 µs		
tx_semaphore_put	0.3 µs		1.3 µs	3.0 µs
tx_mutex_get	0.4 µs	2.4 µs		
tx_mutex_put	1.0 µs		2.0 µs	3.6 µs
tx_event_flags_set	0.6 µs		1.7 µs	3.4 µs
tx_event_flags_get	0.4 µs	2.5 µs		
tx_block_allocate	0.4 µs	2.3 µs		
tx_block_release	0.4 µs		1.4 µs	3.1 µs
tx_byte_allocate	1.4 µs	2.9 µs		
tx_byte_release	0.9 µs		3.0 µs	4.7 µs
Context Switch (CS)	1.2 µs			
Interrupt Latency Range (ILR)	0.0 µs-0.8 µs			

Table 2 – Execution times of various ThreadX services

Efficient interrupt processing delivers the best performance from MicroBlaze processors, giving your product a boost over the competition.

Full Source Code

By providing full source code, you have complete visibility into all ThreadX services – no more mysteries are hidden within the RTOS "black box."

This full view of source code enables you to avoid delays in development caused by an incomplete understanding of RTOS services. With the ThreadX RTOS, full source code is there any time you need it.

Conclusion

Express Logic's ThreadX RTOS is well matched for networking and consumer applications based on MicroBlaze processors. It's been optimized to deliver the processor's inherent performance directly through to the application.

What's next? Express Logic is working to port NetX™, its TCP/IP network stack, to MicroBlaze. With so many of today's consumer electronics devices accessing the Internet, efficient TCP/IP software is increasingly critical. The NetX TCP/IP stack will provide further support for MicroBlaze in consumer electronics products, and will enable developers to concentrate on their application code, getting their product to market faster.

To develop a successful, high-performance consumer, networking, or other electronic product around MicroBlaze processors, use an efficient RTOS like ThreadX by Express Logic. For further information on the ThreadX RTOS and other embedded software products from Express Logic, please visit *www.expresslogic.com*.

Improve Your Productivity with the Embedded Development Kit v6.1i

The new version of EDK offers automatic generation of complex implementation details, new simulation capabilities, and new IP cores.

by Ravi Pragasam
Marketing Manager, Embedded Processor Solutions,
Product Solutions Marketing
Xilinx, Inc.
ravi.pragasam@xilinx.com

The current market environment requires designs to be completed in a short time frame, with features that distinguish them from other products available in the market. Yet the majority of embedded designs fail to meet original specification goals because of time constraints and feature shortcomings in the chosen solution. Often, functionality is compromised to meet performance or quality requirements.

Over the years, embedded system designers had to use standard off-the-shelf discrete components in their applications. Much design time was spent on integrating these standard components so that they met system performance.

ASICs or ASSPs are also chosen as the solution when the production volumes warrant this path. In traditional ASIC flows, designers make most changes in the software to account for hardware limitations and inaccuracies, when problems are discovered late in the design cycle.

This situation is exacerbated as physical geometries shrink and chip fabrication becomes variable from design-to-design, not just from process-to-process. ASSPs are usually standard solutions with some non-standard sections that allow for customization. This customization can set the product apart from other solutions in a certain application area. However, ASSP solutions are also not a perfect fit, as designers have to create workarounds to address feature shortcomings. With standard off-the-shelf discrete components, ASICs and ASSPs, designers end up with solutions that never quite meet their needs.

Xilinx Embedded Processor Solutions

The advent of the programmable platform – with system-level features such as memory, application-specific input/output to support varying interface standards, and now hard- and soft-processor cores – means that integration is already available. With this new capability, you can spend more time on other system challenges, meeting performance goals and shrinking market windows.

Xilinx offers several options to benefit embedded designers. These solutions include both hard- and soft-processor cores – configurable IP cores and a feature-rich design environment – currently available with the Embedded Development Kit (EDK) v6.1i. EDK can help you deliver a highly flexible and feature-rich embedded system.

The hard processor core is an IBM™ PowerPC™ 405 immersed in the Xilinx Virtex-II Pro™ FPGA, delivering 600 D-MIPS at 400 Mhz, with as many as four cores available in the largest Virtex-II Pro device.

Alternatively, Xilinx also offers the MicroBlaze™ 32-bit RISC processor core, delivering up to 150 D-MIPS at up to 125 MHz when used in the Virtex-II Pro device. You can use the MicroBlaze processor to implement low-cost embedded systems, especially when using the Xilinx Spartan™-3 FPGA (see sidebar).

Both soft- and hard-processor cores use IBM CoreConnect bus technology as the interconnect bus for the embedded system. Xilinx also offers a complete set of soft peripherals to implement your embedded system, which is available with the Embedded Development Kit. Examples of these peripheral IP cores include UARTs, GPIO, 10/100 Ethernet MAC, IIC, and high-level data link controller (HDLC) cores that you can use to design embedded solutions for such market segments as networking and communications.

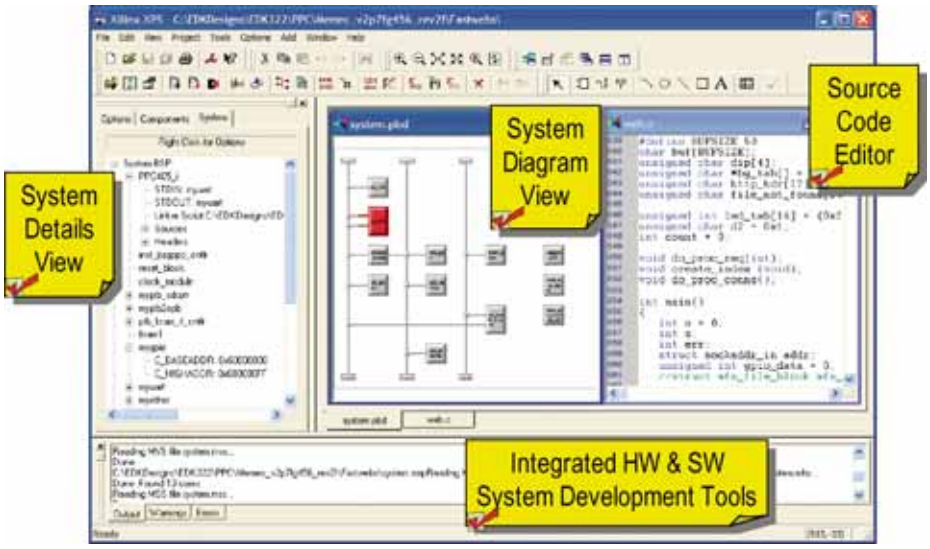


Figure 1 – Xilinx Platform Studio programmable systems integrated design environment

EDK v6.1i

EDK v6.1i is a comprehensive suite of system-level design tools that enables you to design embedded processor systems using Xilinx FPGAs. With the embedded tools within EDK, Xilinx is enabling the adoption of an open format called the Platform Specification Format (PSF). PSF provides an abstraction layer between the hardware and software platforms of an embedded design so that they can be tightly integrated during the development process. This coupling ensures that you can match the programming environment and capability of the software to the hardware resources available.

EDK v6.1i includes the Xilinx Platform Studio (XPS) (Figure 1), an all-encompassing design environment used to define, configure, and generate a custom hardware and matching software and programming environment for either a stand-alone or real-time operating system (RTOS)-enabled programmable system.

Hardware, software, and firmware engineers who need to work in both domains can use the XPS Integrated Design Environment included in EDK. EDK also interfaces with industry-standard software tools from leaders such as Wind River Systems™, MontaVista® Linux®, and popular GNU embedded tools.

Unrivaled Ease of Use

One of the greatest advantages EDK v6.1i offers is the ability to design a customized bootable embedded system in minutes. Using EDK, you can select a target processor core such as the MicroBlaze processor or PowerPC; a specific FPGA architecture; and a list of peripherals to generate a complete custom processing system.

Another huge benefit is the interface to industry-standard tools such as Wind River Systems' Tornado™, which allows the design of high-performance PowerPC-based Virtex-II Pro-centric embedded systems requiring an RTOS.

Xilinx also provides a micro kernel OS within EDK. This provides a small memory footprint that includes several OS-like functions you can use to design PowerPC- or MicroBlaze-based embedded systems.

One of the biggest advantages in EDK that helps you start designing an embedded system in Xilinx FPGAs is the Base System

Builder wizard, shown in Figure 2. This feature within XPS enables the creation of a custom PowerPC- or MicroBlaze-based computing platform (including peripherals and a memory map) with just a few clicks of the mouse. All detailed connections and a default memory map are generated automatically, resulting in a functional hardware design that is ready to be downloaded into a target board.

The automatic generation of a ready-to-use board support package (BSP) matching the defined hardware platform provides tremendous savings in development time. Hardware/software sections of the embedded design that can be problematic are handled automatically by the tools. This dramatically reduces the time-consuming and frustrating phase of the debug and verification cycle common to many embedded applications.

Figure 2 – The Base System Builder wizard enables easy definition of hardware platforms in FPGA-based embedded systems.

Peripheral IP Catalog

The Embedded Development Kit offers an extensive list of soft IP cores that you can use to develop and design embedded systems. The cores are essentially divided into four categories: interface, memory controllers, infrastructure, and peripherals.

Examples of infrastructure cores include bus bridges and arbiters for different CoreConnect buses. Examples of interface IP cores include memory controllers such as Flash, DDR, SDRAM, and SDR that allow the use of external memory controllers as part of the embedded systems. The design environment also allows you to integrate proprietary cores into your design. Also available for your use is an extensive list of general peripheral cores such as SPI, IIC, UART, Timer/Counter, Watch Dog Timer, and GPIO, found in any processor subsystem.

There is also a growing list of high-value IP cores that you can purchase independently to plug-and-play in EDK. Examples of these high-value cores include Ethernet MAC 10/1000, 1 Gigabit Ethernet, and the HDLC controller with as many as 256 channels. The biggest advantage of these cores is that most of them are parameterizable within EDK. As opposed to standard solutions where not all of the features may be required for a design, embedded systems using these cores use only the logic that is needed without any waste.

Xilinx also has several partners that offer IP cores you can use to develop embedded designs in FPGAs.

Integration with ISE Logic Design Tools

Yet another benefit is the integration of the Embedded Development Kit with the recently released Xilinx ISE 6.1i. You can now access EDK from within ISE, thus eliminating the need to leave the ISE environment, and design a complete embedded system by invoking the EDK tools within ISE. This integration with Xilinx ISE 6.1i enables a more seamless and intuitive development environment with the rest of your FPGA design.

Debug Options

The Embedded Development Kit provides a feature-rich environment to debug designs in software and hardware. It includes a PowerPC and MicroBlaze Instruction Set Simulator (ISS) to enable the debugging of software code on the development workstation. Hardware simulation has been enhanced, with the ability to do mixed VHDL/Verilog™ simulation (including Xilinx LogiCORE™ and customer-created IP) as well as the ability to toggle between real-time logic (RTL), structural, and full timing-based simulations. The Single Step debugger from WindRiver Systems is also available for PowerPC in Virtex-II Pro-based embedded systems.

Extending the Xilinx leadership in hardware on-chip debug, EDK version 6.1i enables on-chip cross triggering between the logic and bus analyzer cores available with the Xilinx ChipScope™ Pro software tool and the GNU (gdb) debugger. Support for the visionProbe hardware debugger from WindRiver Systems is also available for PowerPC in Virtex-II Pro-based embedded systems.

Conclusion

Traditional embedded designers who had to use standard discrete off-the-shelf components, ASICs, or ASSPs to satisfy their system application needs now have the option to use a programmable platform to develop a custom system. Using a programmable platform offers several advantages, such as customization, consolidation, integration, and protection from component obsolescence.

The ability to use a design environment that provides an all-encompassing suite of tools and interfaces to familiar design entry methods further facilitates the development process and increases the overall speed of the design process. With EDK v6.1i, you now have access to a suite of embedded tools from Xilinx that take care of those complicated steps that are part of the embedded design process. Thus, you can focus more on creating a custom platform with unique features that bring value to your system, as opposed to the mundane ones available from your competitors. Σ

MicroBlaze in Spartan-3 Delivers Soft Processor for Less than 75 Cents

by Helen Yu
Embedded Processor Solutions, Product Solutions Marketing
Xilinx, Inc.
helen.yu@xilinx.com

The MicroBlaze™ soft-core processor in Spartan™-3 FPGAs provides an attractive solution for low-cost processing with its lower price points. You can now implement the 32-bit MicroBlaze processor in the Spartan-3 XC3S1000 device for an effective price of $0.73*.

Additionally, you can implement the Xilinx 8-bit PicoBlaze™ soft processor in the same device for an effective price of $0.13*. These new price points offer a significant cost savings that is comparable to stand-alone discrete solutions available today.

The new breakthrough price points for the Spartan-3 FPGA family are due to the successful implementation of 90 nm technology. With new architecture optimized for 90 nm technology, Xilinx has packed more logic cells onto smaller die, resulting in a cost-per-logic-cell advantage as high as 36% over competing solutions.

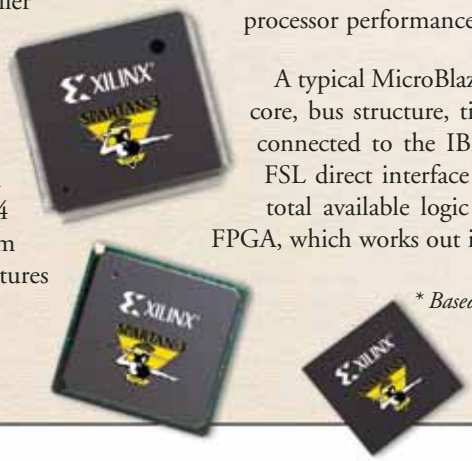
You can also take advantage of the new features in the just-released Embedded Development Kit v6.1i (such as the Base System Builder and IP import wizard) to design a complete MicroBlaze-based system in a relatively short time, thus increasing your overall productivity and meeting time-to-market requirements.

The MicroBlaze processor is the industry's most popular soft 32-bit processor, with performance up to 68 D-MIPS when implemented in the Spartan-3 fabric. Combined with soft peripherals and logic, it can address many traditional 16- and 32-bit microprocessor and microcontroller applications, such as process control instrumentation, test and measurement, automotive, and high-end consumer.

The easy-to-use and popular PicoBlaze soft processor delivers up to 40 MIPS when implemented in the Spartan-3 fabric and can be used as a simple microcontroller in various applications.

More Features for Less Cost

Spartan-3 FPGAs offer a combination of features and density, along with large embedded block and distributed RAM, and as many as 784 I/Os. This makes the devices ideal for system integration and the addition of unique new features during the product's life cycle.



The MicroBlaze 32-bit RISC processor core, along with a standard set of soft peripherals, is included with the Embedded Development Kit. Advanced features in the MicroBlaze processor that can help embedded designers meet their performance needs include:

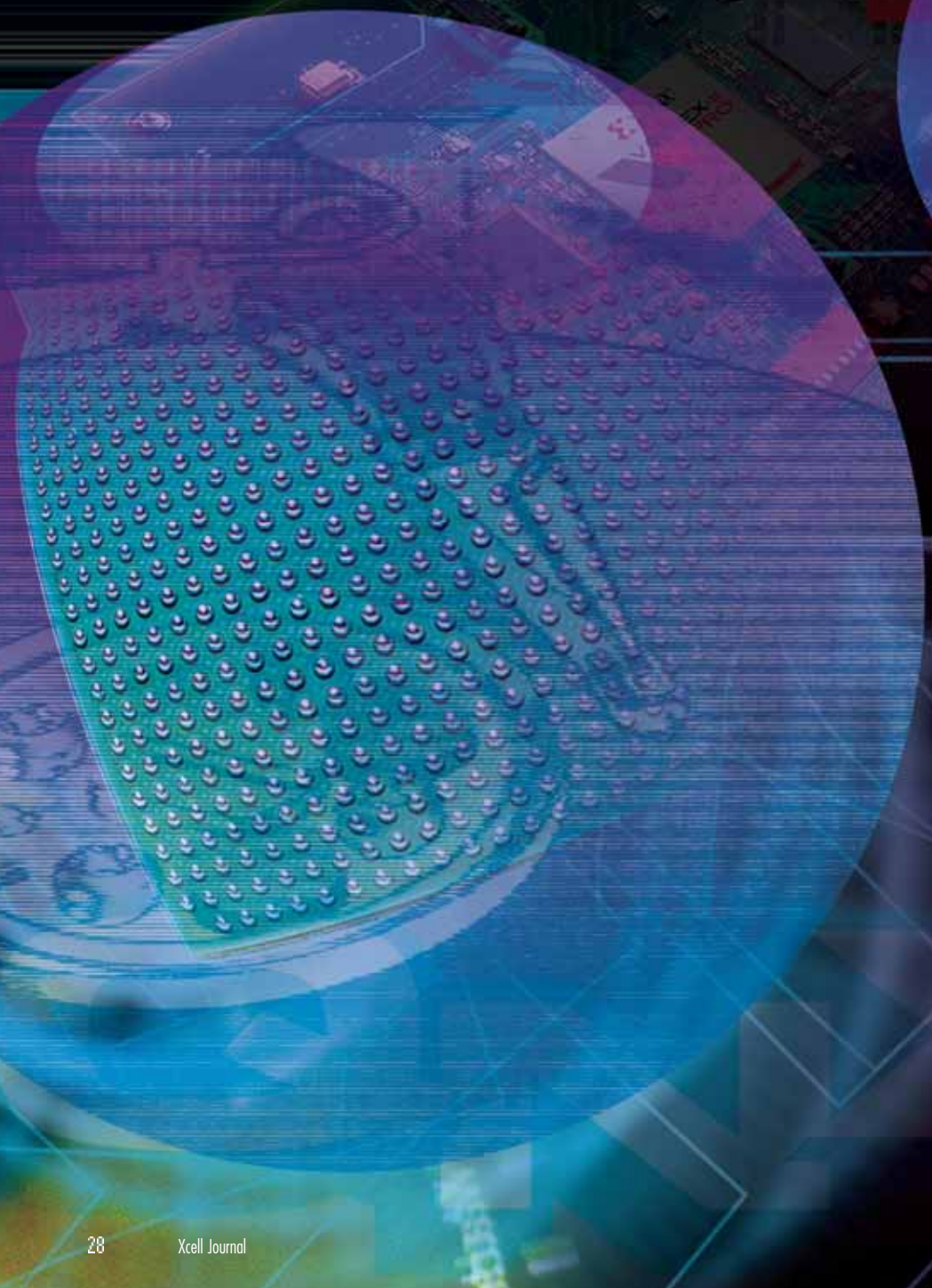
- User-configurable caches. The MicroBlaze soft processor includes direct-mapped, level one (L1) instruction and data caches that are configurable up to 64 KB. Memory is divided by address to provide 1 GB of cacheable memory space and 3 GB of non-cacheable memory space to significantly reduce instruction execution time and code space.
- Fast simplex link (FSL) to simplify software-to-hardware implementations. FSL offers a 300 MB/s direct processor interface and point-to-point connection for custom functions and hardware. This high-bandwidth, configurable depth first-in first-out (FIFO) interface to the CPU is ideal for streaming applications. Each of the 32 input/output FSL connections uses SRL16 for the FIFO, which consumes only 36 look-up tables for a 32-wide, 16-deep FIFO.
- Hardware debug module. The MicroBlaze debug module offers JTAG support for multi-processor debugging. Debug options include configurable break points and watch points, non-intrusive debugging, and debugging of ROM code – all available as part of the Embedded Development Kit.
- 32-bit barrel shifter. The 32-bit barrel shifter is particularly advantageous for applications that shift data logic left or right – a normal code style for C programs. Independent of the shift amount or direction, barrel shifter instructions take only two clock cycles and can boost shift execution up to 15 times.
- Hardware divide. Xilinx has implemented a dedicated MicroBlaze instruction with hardware support for divide functions; you no longer need to run a software library function to perform divides. The hardware divider, combined with other enhanced features, results in higher processor performance.

A typical MicroBlaze system including the soft processor core, bus structure, timer, GPIO, and UART peripherals connected to the IBM™ CoreConnect bus or the new FSL direct interface consumes approximately 8% of the total available logic resources in the 3S1000 Spartan-3 FPGA, which works out in cost to approximately $1.01*.

** Based on volume pricing for 250K units, 2004*

QNX Neutrino RTOS Optimizes Programmable Systems

QNX support for Virtex-II Pro devices enables software upgrades and fault tolerance before, during, and after deployment.



by Paul Leroux
Technology Analyst
QNX Software Systems Ltd.
paull@qnx.com

Who would have thought RAM and ROM memory, digital signaling processing, customizable IP, embedded hard- and soft-core microprocessors, multi-gigabit serial transceivers, controlled impedance technology, and remote-reconfigurable FPGA logic fabric could all be integrated into a single, reprogrammable device?

That's exactly the level of integration offered by system-on-chip (SOC) platforms like the Xilinx Virtex-II Pro™ FPGA. This device not only reduces board size and simplifies the manufacturing process, but is programmable (fast to start), reprogrammable (quick to fix or enhance), and field-upgradable (possible to upgrade or optimize after deployment). Moreover, it comes in a variety of densities, packages, and configurations, allowing you to deploy an SOC tailored to your specific requirements.

To exploit these numerous capabilities, you need appropriate tools and operating system technology. Consequently, Xilinx has worked with QNX Software Systems to provide an integrated development environment (QNX® Momentics® development suite) and real-time operating system (QNX Neutrino® RTOS) for the IBM PowerPC™ 405 processors embedded in Virtex-II Pro FPGAs. Together, these QNX products offer software upgradability, fault tolerance, and true real-time performance – along with a rich complement of tools for building, debugging, and fine-tuning software applications.

Why Use an RTOS?

Before taking a closer look at the QNX Neutrino RTOS, let's examine why your programmable system would even need an RTOS in the first place. The answer, as we'll see, goes beyond issues of sheer speed or efficiency.

On any given day, you may use dozens of embedded applications. In some cases, you probably don't mind if the application takes a second or two to respond. Waiting for an ATM machine to dispense cash is a perfect example.

An ATM represents the world of "soft" real time, where a delay might diminish the value of a system operation, but it doesn't pose a safety hazard, cause a major inconvenience, or lead to an economic penalty. An RTOS, with its ability to guarantee fast response times, can be beneficial for such applications, but it is not essential.

In many applications, however, even a slight delay or timing error can constitute system failure. Consider a router handling VoIP where delayed packets can result in unacceptable voice jitter, or an in-car telematics system where consumers won't tolerate a user interface that fails to respond immediately to input.

Similar delays can cause the effective failure of medical monitoring devices, consumer appliances, and a variety of other software-dependent products. Such systems – often dubbed "hard" real-time systems – need an OS like the QNX Neutrino RTOS to deliver the predictable response times demanded of them.

How does an RTOS enable this predictability? To begin with, it ensures that software processes always execute in order of their priority. If a high-priority, time-critical process becomes ready to run, the RTOS will immediately take over the CPU from any lower-priority process. Moreover, the high-priority process can run uninterrupted until it has finished what it needs to do – unless, of course, it is preempted by a process with an even higher priority.

This approach, known as "preemptive scheduling," helps ensure that time-critical processes meet their deadlines consistently, regardless of how many other processes are competing for CPU time.

Besides preemptive scheduling, the QNX Neutrino RTOS provides a number of other mechanisms and capabilities to enable fast, predictable response to critical events. These include support for:

- Microsecond and even sub-microsecond interrupt latencies
- Scheduling methods that set a capped limit on the execution time of threads
- Nested interrupts that ensure the RTOS always handles high-priority interrupts first.

The QNX Neutrino RTOS also supports "distributed priority inheritance," which enables any OS (for instance, a device driver) to execute at the priority of the application requesting the service. This ensures that work done for a low-priority application doesn't prevent a higher-priority process from accessing the CPU – a condition known as "priority inversion."

Nonetheless, RTOSs aren't just for hard real-time systems. By providing precise control over the timing of every application and system service, the QNX Neutrino RTOS can make it much easier for you to optimize the performance and behavior of virtually any embedded design.

Engineered for Upgradable, Reconfigurable Systems

Although Virtex-II Pro FPGAs provide a platform for creating upgradable hardware solutions, few RTOSs offer the equivalent upgradability on the software side. For instance, in a conventional RTOS, most software components – the OS kernel, networking stacks, drivers, and applications – run together in a single memory address space. This architecture is efficient, but has two immediate drawbacks:

1. A memory violation in any component can corrupt the OS kernel or any other component, causing system-wide failure.
2. It is difficult, if not impossible, to dynamically add, repair, or replace individual components.

The QNX Neutrino RTOS takes a very different approach. Rather than pack software components into a single, unwieldy "monolith," it allows applications, drivers, file systems, and networking stacks to run in separate, memory-protected address spaces (Figure 1).

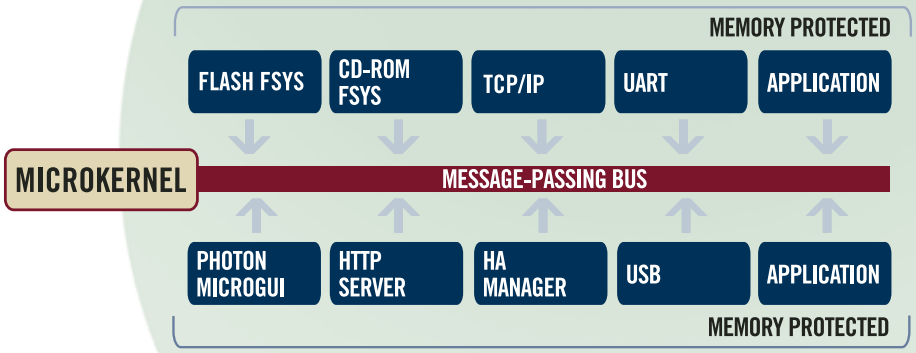


Figure 1 – In the QNX Neutrino RTOS, you can dynamically plug in or plug out software components as needed.

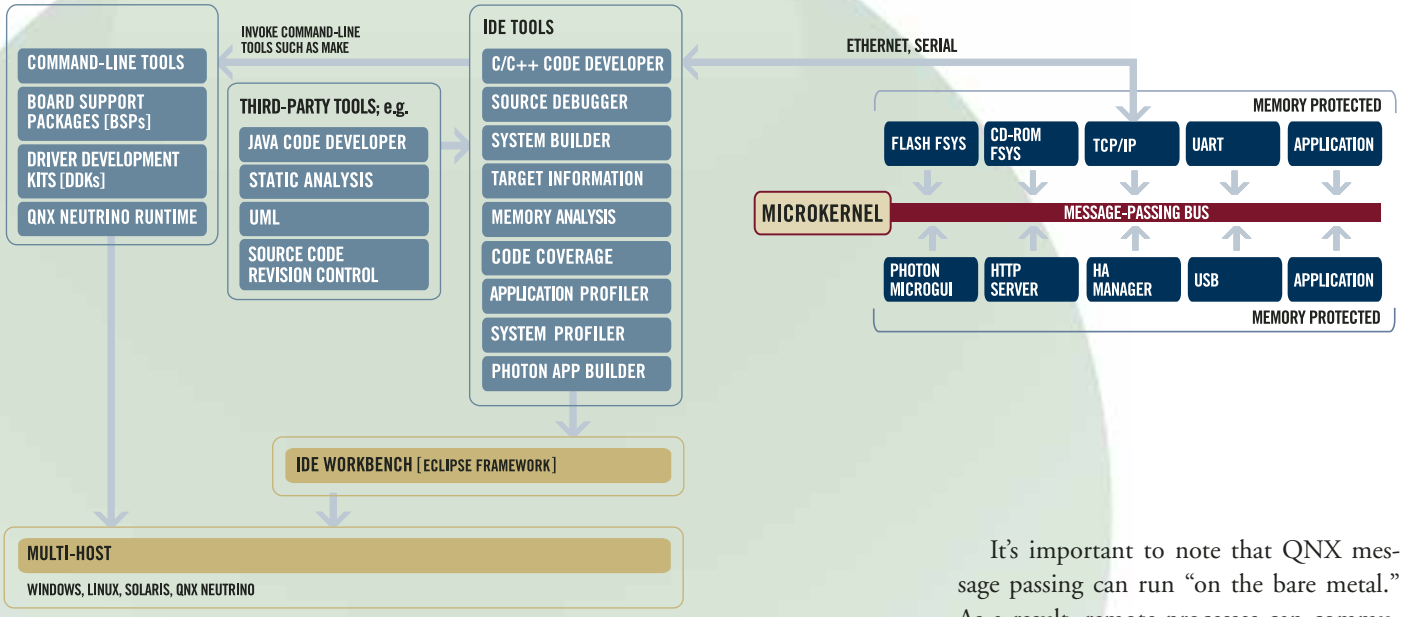


Figure 2 – The QNX Momentics suite supports multiple languages and development hosts.

This approach offers several benefits:

- A fault in any software component won't damage the OS kernel or any other component.
- Any component that commits a memory or logic error can be automatically restarted while the rest of the system continues to run.
- Virtually any component, even a low-level driver, can be started or replaced on the fly, without rebooting.

Together, these capabilities make the QNX Neutrino RTOS an ideal OS for high-availability systems (such as network elements, in-car telematics units, or TV broadcast systems) that must run nonstop – even if individual components fail or hardware subsystems need to be replaced or reconfigured. In fact, QNX-based systems have a reputation for running 10 years or longer without downtime.

The modularity of the QNX Neutrino RTOS also makes it ideal for resource-constrained systems, as it allows a system to run only the software components it currently requires. Any component that isn't required can be dynamically removed and then restarted when the need arises.

Distributed Processing for Multiprocessor Platforms

As Figure 1 illustrates, all software components in the QNX Neutrino RTOS communicate via a single, well-defined communication mechanism: high-speed message passing. This message passing forms a virtual "software bus" that lets you plug in – or plug out – any component independently of other components, much like the hardware buses of modern networking equipment.

QNX message passing has other benefits as well. Messages can flow transparently across processor boundaries, allowing applications to seamlessly access services running on remote nodes.

This transparency offers interesting possibilities for Virtex-II Pro FPGA-based systems incorporating multiple PowerPC 405 embedded processors. For example, an application running on one processor could access a device driver or communications stack running on another processor as if that service was on the local processor. No special code or programming is required. In addition, QNX message passing automatically synchronizes the execution of the application and the remote service, even though they reside on different CPUs.

It's important to note that QNX message passing can run "on the bare metal." As a result, remote processes can communicate with each other without the overhead of a conventional networking protocol, such as TCP/IP (although TCP/IP can be used if desired).

In addition, QNX message passing doesn't require any special programming skills or application programming interfaces (APIs). To exchange messages, processes can simply use industry-standard Portable Operating System Interface (POSIX) calls, such as open(), read(), write(), and lseek(). The actual job of passing messages between processes is handled transparently by the underlying C library.

Standard Interfaces Ease Portability

Speaking of POSIX, the QNX Neutrino RTOS complies with POSIX 1003.1-2001 API standards, including threads, real-time extensions, and a number of other options. Consequently, it is quite easy to port Linux™, Unix™, and other open-source applications; in most cases, you simply recompile and relink the source. Programmers with Linux or Unix experience can become productive almost immediately in the QNX Neutrino operating environment.

Validated Support for Platform IP

The QNX Neutrino RTOS has been ported to the IBM PowerPC 405 embedded processor in the Virtex-II Pro FPGA. In addition, QNX Software Systems has port-

ed (and validated with Xilinx) support for customizable platform IP cores such as bootloader and startup code, as well as drivers for the universal asynchronous receiver transmitter (UART), interrupt controller (INTC), 10/100 Ethernet MAC, and I^2C. The Xilinx ML300 reference platform with Virtex-II Pro FPGA and PowerPC 405 processor served as the target board for this process.

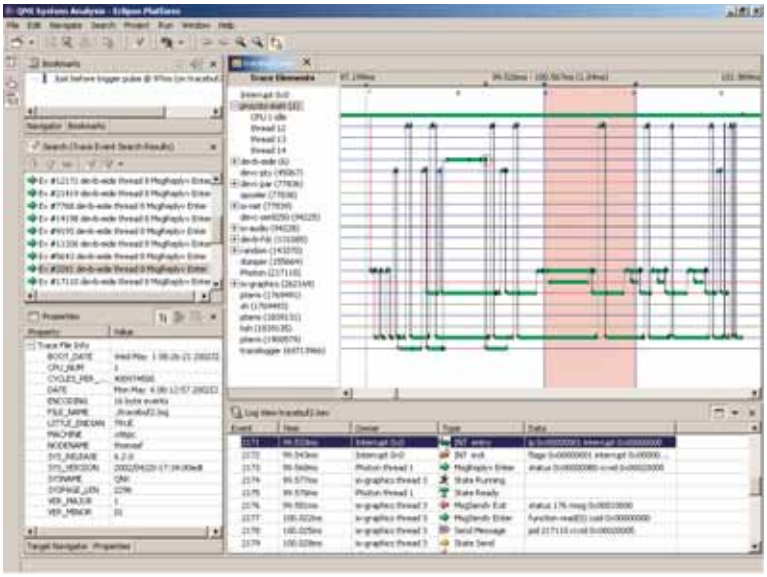


Figure 3 – Diagnostic tools like the system profiler help optimize performance.

Integrated Toolset Optimizes Workflow

QNX Software Systems has also extended its QNX Momentics development suite to target the PowerPC processor in the Virtex-II Pro FPGA. The QNX Momentics suite supports all the tools you might expect, such as graphical code editors, debuggers, compilers, and project builders. As Figure 2 illustrates, it also includes a complement of non-intrusive diagnostic tools to fine-tune applications on your programmable system.

The integrated development environment (IDE) in the QNX Momentics suite is extremely flexible. You can:

- Code in multiple languages (C, C++, Embedded C++, Java).
- Develop on several host environments (Windows™, Solaris™, Linux, QNX Neutrino RTOS).
- Use a mix of source-control protocols
- Choose between your own build commands or the IDE's multi-CPU makefile framework.

The QNX Momentics suite also supports industry-standard tools such as the GNU GCC compiler and GDB debugger, as well as a graphical source debugger that is tightly integrated with the code editors, memory analysis plug-in, and other tools.

Developers must focus on writing reliable software in the least amount of time. As a result, the QNX Momentics suite includes wizards to automate mundane tasks, such as creating new projects, configuring remote debug sessions, and building custom flash file system images for target hardware. Plus, all tools – even third-party plug-ins – share the same user interface, making the IDE easy to learn.

To simplify diagnosis of system behavior, the QNX Momentics development suite offers several visualization tools:

- The application profiler supports both statistical and instrumented profiling to pinpoint inefficient code and algorithms. It can drill down to the source-line level, showing which lines are consuming the most processor cycles.
- The memory analysis plug-in graphically displays memory usage to help catch overruns, underruns, invalid use of memory, and freeing the same memory twice.
- The system profiler performs trace analysis to revolve timing conflicts, pinpoint deadlocks, root out logic flaws, detect hidden faults, and optimize system-level performance. For an example of the system profiler in action, see Figure 3.

These tools are engineered to be non-invasive, allowing you to accurately troubleshoot systems both in the lab and in the field.

To simplify integration of third-party tools, the QNX Momentics development suite is based on the Eclipse universal tool platform. The Eclipse platform is supported by a large and rapidly growing community of tool vendors and developers. Eclipse provides well-defined interfaces to ensure that tools – even those from multiple vendors – work together seamlessly. Eclipse also allows previously isolated or unextensible tools to share data with each other, without the need for manual intervention. This integration can improve workflow immensely, allowing you to quickly transition from one task (stepping through the start-up code for an embedded device) to another (debugging applications running on the embedded device).

Conclusion

To achieve the right mix of hardware, software, and embedded operating system technology in a programmable platform like the Virtex-II Pro FPGA, you must choose system components and development tools wisely. The QNX Neutrino RTOS allows your application to gain the benefits of software reliability and hard real-time performance, and it lets you scale OS features to the size and nature of the design.

Using the QNX Momentics IDE introduces the flexibility of choosing your preferred host and language, as well as command-line or graphic interaction. The highly integrated architecture of the IDE can streamline your development process, while the non-invasive diagnostic tools let you quickly fine-tune your final product.

To learn more about the QNX Neutrino RTOS, the QNX Momentics development suite, and their support for the Virtex-II Pro FPGA, visit QNX Software Systems at *www.qnx.com.* Σ

Wind River Delivers Embedded System Performance

Create efficient, speedy code with the Diab XE C/C++ Compiler Suite for Xilinx FPGAs.

by Kent Shaw
Technical Marketing Engineer
Wind River Systems
kent.shaw@windriver.com

All compilers share many common features. However, compilers designed specifically for embedded systems have several unique requirements:

- Flexibility and control
- Strict adherence to standards
- Industry-leading optimization technology.

The Diab™ C/C++ Compiler Suite from Wind River Systems® has gained an industry-wide reputation for generating the fastest and smallest code. However, it takes more than performance to become the leading embedded compiler.

In this article, we will introduce the Diab XE (Xilinx Edition) embedded compiler from Wind River and explain how it can help you develop a superior embedded processing system with Xilinx Virtex-II Pro™ FPGAs and IBM PowerPC™ (PPC)-enabled devices.

Adherence to Standards

Standards are critical for portability and interoperability with other tools. The Diab XE compiler uses the latest version of the EDG (Edison Design Group) C++ front end and is 100% compliant with American National Standards Institute (ANSI) C++ standards. The Diab XE compiler also uses run-time libraries from Dinkumware™ Ltd., which provides 100% ANSI compliance for run-time libraries and the Standard Template Library (STL).

For applications where code size is critical, the Diab XE compiler provides an abridged version of the C++ run-time library. This version of the library is also 100% standards-complaint, but it is built with exception processing disabled, thereby generating much smaller applications. Tests have shown this library to result in code sizes as much as 38% smaller.

Another aspect of standards compliance is output formats. The Diab XE compiler maintains strict compliance with the various embedded application binary interfaces (EABIs). Wind River also works with embedded processor vendors to stay current and is actively involved with the development of EABIs.

Adherence to standards leads to easier porting and ensures interoperability with other development and maintenance tools.

Modular Architecture

As shown in Figure 1, the Diab XE compiler is based on a modular design comprising three primary modules:

- The front-end: Source code language is parsed into tokens. These tokens are the same regardless of the language used – C, C++, Java, and others.
- The optimization stage: Optimizations are applied to all languages, because the tokens generated by the front-end are the same regardless of language.
- The back-end: Code is generated. This stage is target-specific. Diab XE's modular design permits the compiler to fully utilize the architecture of the IBM PowerPC 405 microprocessors within Xilinx FPGAs.

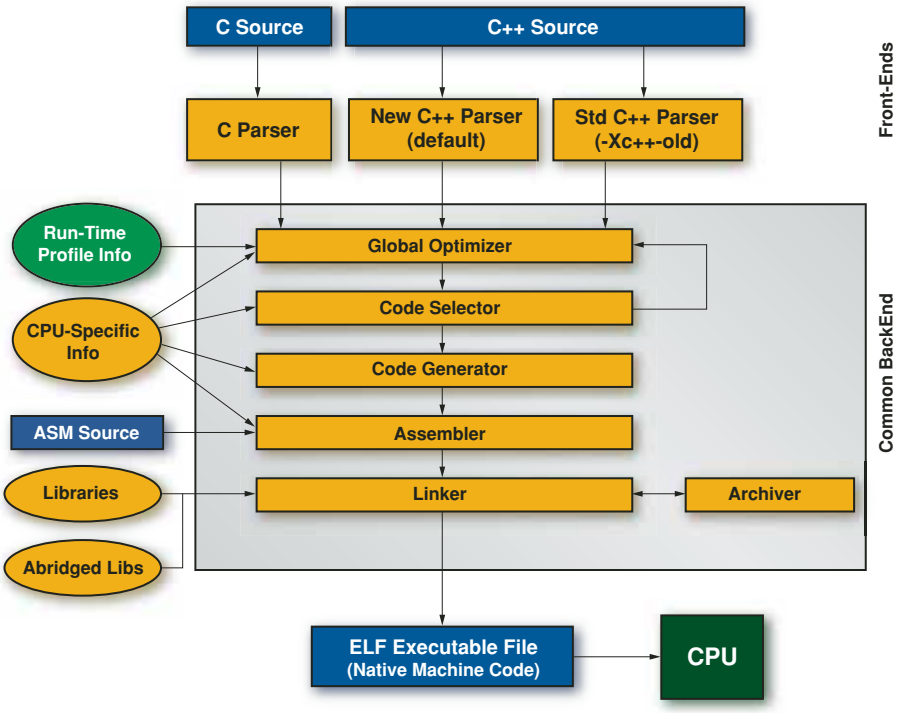


Figure 1 – Diab modular architecture

Diab XE's modular design enables you to take advantage of all non-target-specific optimizations. As the front-end and optimization stages are shared, all advancements made in these stages apply to supported Diab XE target architectures.

Optimization Technology

Optimization falls into two primary areas – front-end and back-end. Front-end optimizations are concerned with the language. Back-end optimizations use architecture-specific code generation to use the design aspects of the particular target processor.

Although almost all compilers do a good job with front-end optimizations, the Diab tool suite distinguishes itself from other compilers with its efficient back-end optimization technology.

The growing hardware complexity of today's processor designs makes it very difficult to manually fine-tune applications. The Diab XE compiler is specifically engineered to enable you to maximize pipelines, caches, and other architecture-specific capabilities, including the IBM PowerPC 405 cores in Xilinx FPGAs.

The Diab XE compiler can further optimize your application code based on application profile information. No more manual tweaking of your code is required – the Diab XE compiler will do this for you.

Because Wind River Systems owns and controls the technology that drives the Diab XE compiler, they can advance the technology on their own schedule without requiring approval from an outside source.

Flexibility and Control

Embedded systems typically have many design constraints – limited memory, limited performance. To meet the demands imposed by these constraints, an effective embedded compiler must provide a wide range of flexibility and control, with features such as:

- Addressing modes, which can have a direct impact on both performance and code size. The Diab XE compiler uses intelligent default settings, but it provides options to give you complete control. Separate options also exist for specifying both code and data modes.
- Support for position-independent code and data (PIC/PID).
- Structure and bit-field packing. Here too, intelligent defaults are used, but with options for complete control over how structures are packed and how structure members are aligned. You can also specify the type (short, int, long) used for bit-fields.
- Function inlining, which can result in significant performance gains. Although all compiler vendors make use of this technique, only the Diab XE compiler enables you to specify the precise size of the functions to be inlined, giving you control over the memory footprint.
- Loop unrolling. This is similar to function inlining, because it can result in significant performance gains. Most compiler vendors use loop unrolling. However, only the Diab XE compiler gives you the ability to control both the size of loops to be unrolled and the number of times to unroll them.
- Embedded assembly code. The Diab XE compiler provides two methods: a relatively simple string-based assembly function and a powerful macro-based function. The string-based approach is ideal for writing values to a register. The macro approach provides total flexibility and control over which registers are used for the macros.
- Diab XE's robust linker command files provide the controls for you to precisely place all program elements in the correct memory locations. Because embedded applications often have strict requirements for locating variables, functions, and data in memory, this is an especially important feature when designing systems with multiple memory types and sizes.

Quality Documentation

Diab XE's documentation makes it easy to learn about and take advantage of the compiler's power, flexibility, and control. All

manuals have separate sections devoted to each component in the suite:

- Compiler
- Assembler
- Linker
- Utilities
- Libraries
- ELF/COFF formats.

Brief Comparison with GNU GCC

The 2.96 release of the GNU Compiler Collection (GCC) for C/C++ is a robust and efficient compiler. Yet it lacks some of the features for a truly effective embedded compiler, because it was originally developed for computer systems running the Unix™ operating system.

For example, the GNU GCC C/C++ 2.96 release does not recognize the PowerPC 405 core as a specific target and cannot take advantage of PPC 405-only registers.

When your application needs to be as small and as fast as it can be, the Diab XE compiler will provide you with superior results. The accompanying sidebar "High Performance or Small Footprint?" clearly shows the superiority of the Diab XE C/C++ Compiler 5.6 when compared to the GNU GCC C/C++ 2.96 in four critical parameters:

- Dhrystone performance in milliseconds
- Dhrystone code size
- DES application performance
- DES application size.

Conclusion

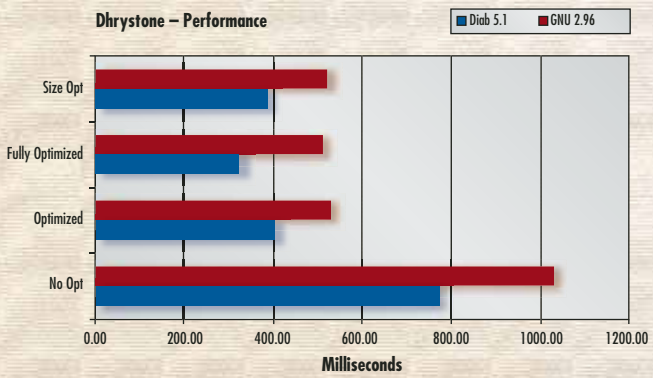
As a core development tool, the Diab XE C/C++ Compiler Suite enables you to create exceptionally fast, tight code for your embedded applications.

The superior flexibility and control afforded by the Diab XE compiler enables developers to fine-tune the compiler's output to meet and exceed the needs of demanding embedded applications.

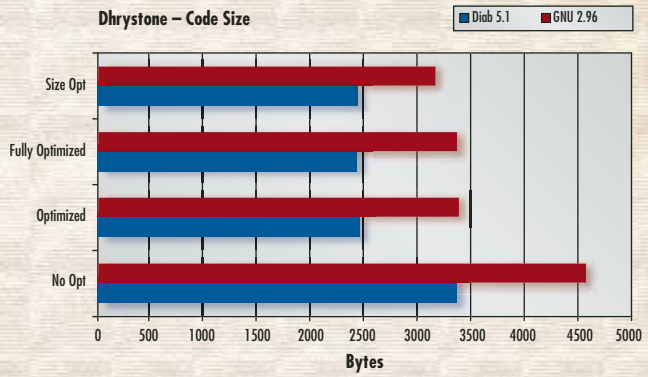
The Diab XE C/C++ Compiler Suite is included in the Wind River package, which is available from Xilinx at *www.xilinx.com* (search for "Diab XE C/C++ Compiler"). This version is full featured but limited to a fully linked object size of 256 KB – enough room to develop small applications and well suited to Virtex-II Pro FPGAs with embedded PowerPC processors.

High Performance or Small Footprint?

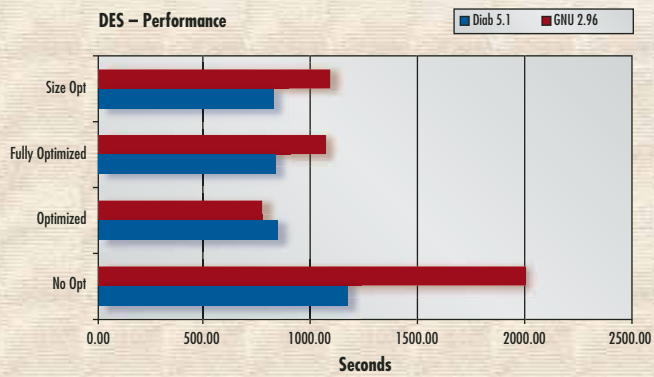
By using the Diab XE C/C++ Compiler Suite from Wind River, you can have it both ways. The four charts below show the increased performance and smaller code size produced by the Diab XE compiler when compared to a GNU compiler. All measurements were made using Diab XE version 5.1 and GCC 2.96.



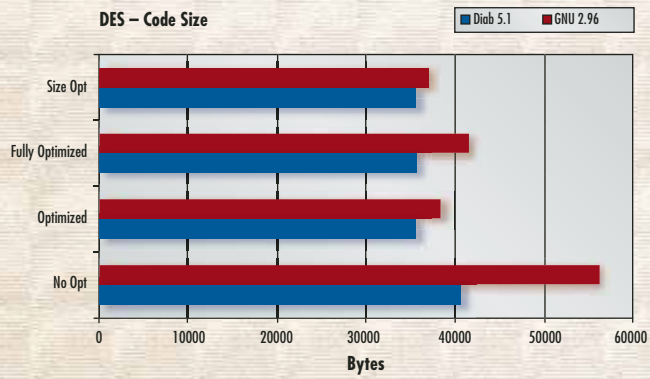
Dhrystone performance comparison



Dhrystone code size comparison



DES application performance comparison



DES application size comparison

Use an RTOS on Your Next MicroBlaze-Based Product

An RTOS called µC/OS-II has been ported to the MicroBlaze soft-core processor.

by Jean J. Labrosse
President
Micrium, Inc.
Jean.Labrosse@Micrium.com

Designing software for a microprocessor-based application can be a challenging undertaking. To reduce the risk and complexity of such a project, it's always important to break the problem into small pieces, or tasks. Each task is therefore responsible for a certain aspect of the application. Keyboard scanning, operator interfaces, display, protocol stacks, reading sensors, performing control functions, updating outputs, and data logging are all examples of tasks that a microprocessor can perform.

In many embedded applications, these tasks are simply executed sequentially by the microprocessor in a giant infinite loop. Unfortunately, these types of systems do not provide the level of responsiveness required by a growing number of real-time applications because the code executes in sequence. Thus, all tasks have virtually the same priority.

For this and other reasons, consider the use of a real-time operating system (RTOS) for your next Xilinx MicroBlaze™ processor-based product. This article introduces you to a low-cost, high-performance, and high-quality RTOS from Micrium called µC/OS-II.

What is µC/OS-II?

µC/OS-II is a highly portable, ROM-able, scalable, preemptive RTOS. The source code for µC/OS-II contains about 5,500 lines of clean ANSI C code. It is included on a CD that accompanies a book fully describing the inner workings of µC/OS-II. The book is called *MicroC/OS-II, The Real-Time Kernel* (ISBN 1-5782-0103-9) and was written by the author (Figure 2). The book also contains more than 50 pages of RTOS basics.

Although the source code for µC/OS-II is provided with the book, µC/OS-II is not freeware, nor is it open source. In fact, you need to license µC/OS-II to use it in actual products.

Micrium's application note AN-1013 provides the complete details about the porting of µC/OS-II to the MicroBlaze soft-core processor. This application note is available from the Micrium website at *www.micrium.com*.

µC/OS-II was designed specifically for embedded applications and thus has a very small footprint. In fact, the footprint for µC/OS-II can be scaled based on which µC/OS-II services you need in your application. On the MicroBlaze processor, µC/OS-II can be scaled (as shown in Table 1) and can easily fit into Xilinx FPGA block RAM.

µC/OS-II is a fully preemptive real-time kernel. This means that µC/OS-II always attempts to run the highest-priority task that is ready to run. Interrupts can suspend the execution of a task and, if a higher-priority task is awakened as a result of the interrupt, that task will run as soon as the interrupt service routines (ISR) are completed.

µC/OS-II provides a number of system services, such as task and time management, semaphores, mutual exclusion semaphores, event flags, message mailboxes, message queues, and fixed-sized memory partitions. In all, µC/OS-II provides more than 65 functions you can call from your application. µC/OS-II can manage as many as 63 application tasks that could result in quite complex systems.

µC/OS-II is used in hundreds of products from companies worldwide. µC/OS-II is also very popular among colleges and universities, and is the foundation for many courses about real-time systems.

	Code (ROM)	Data (RAM)
Minimum Size	5.5 Kb	1.25 Kb (excluding task stacks)
Maximum Size	24 Kb	9 Kb (excluding task stacks)

Table 1 – Memory footprint for µC/OS-II on the MicroBlaze processor

RTOS Basics with µC/OS-II

When you use an RTOS, very little has to change in the way you design your product; you still need to break your application into tasks. An RTOS is software that manages and prioritizes these tasks based on your input. The RTOS takes the most important task that is ready to run and executes it on the microprocessor.

With most RTOSs, a task is an infinite loop, as shown here:

```
void MyTask (void)
{
  while (1) {
    // Do something (Your code)
    // Wait for an event, either:
    //  Time to expire
    //  A signal from an ISR
    //  A message from another task
    //  Other
  }
}
```

You're probably wondering how it's possible to have multiple infinite loops on a single processor, as well as how the CPU goes from one infinite loop to the next. The RTOS takes care of that for you by suspending and resuming tasks based on events. To have the RTOS manage these tasks, you need to provide at least the following information to an RTOS:

- The starting address of the task (MyTask())
- The task priority based on the importance you give to the task
- The amount of stack space needed by the task.

With µC/OS-II, you provide this information by calling a function to "create a task" (OSTaskCreate()). µC/OS-II maintains a list of all tasks that have been created and organizes them by priority. The task with the highest priority gets to execute its code on the MicroBlaze processor. You determine how much stack space each task gets based on the amount of space needed by the task for local variables, function calls, and ISRs.

In the task body, within the infinite loop, you need to include calls to RTOS functions so that your task waits for some event. One task can ask the RTOS to "run me every 100 milliseconds." Another task can say, "I want to wait for a character to be received on a serial port." Yet another task can say, "I want to wait for an analog-to-digital conversion to complete."

Events are typically generated by your application code, either from ISRs or issued by other tasks. In other words, an ISR or another task can send a message to a task to wake that task up. With µC/OS-II, a task waits for an event using a "pend" call. An event is signaled using a "post" call, as shown in the pseudo-code below:

```
void EthernetRxISR(void)
{
  Get buffer to store received packet;
  Copy NIC packet to buffer;
  Call OSQPost() to send packet to task
  (this signals the task);
}

void EthernetRxTask (void)
{
  while (1) {
    Wait for packet by calling OSQPend();
    Process the packet received;
  }
}
```

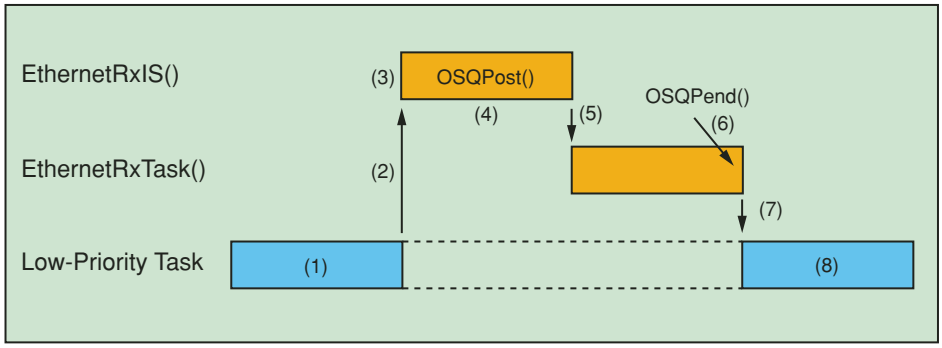


Figure 1 – Execution profile of an ISR and µC/OS-II

The execution profile for these two functions is shown in Figure 1.

Let's assume that a low-priority task is currently executing (1). When an Ethernet packet is received on the Network Interface Card (NIC), an interrupt is generated. The MicroBlaze processor suspends execution of the current task (the low-priority task) and vectors to the ISR handler (2).

With µC/OS-II, the ISR starts by saving all of the MicroBlaze registers, taking a mere 300 nanoseconds at 150 MHz. The ISR handler then needs to determine the source of the interrupt and execute the appropriate function (in this case, EthernetRxISR()) (3).

This ISR obtains a buffer large enough to hold the packet received by the NIC. It copies the contents of the packet received into the buffer and sends the address of this buffer to the task responsible for handling received packets (EthernetRxTask() in our example) using µC/OS-II's OSQPost() (4).

When this operation is completed, the ISR invokes µC/OS-II (calls a function) and µC/OS-II determines that a higher-priority task needs to execute. µC/OS-II then resumes the more important task and doesn't return to the interrupted task (5). With µC/OS-II on the MicroBlaze processor, this takes about 750 nanoseconds at 150 MHz.

When the packet is processed, EthernetRxTask() calls OSQPend() (6). µC/OS-II notices that there are no more packets to process and suspends execution of this task by saving the contents of all MicroBlaze registers onto that task's stack. Note that when the task is suspended, it doesn't consume any processing time.

µC/OS-II then locates the next most important task that's ready to run (the "Low-Priority Task" in Figure 1) and switches to that task by loading the MicroBlaze registers with the context saved by the ISR (7). This is called a *context switch*. µC/OS-II takes less than 1 microsecond at 150 MHz on the MicroBlaze processor to complete this process. The interrupted task is resumed exactly as if it was never interrupted (8).

The general rule for an application using an RTOS is to put most of your code into tasks and very little code into ISRs. In fact, you always want to keep your ISRs as short as possible. With an RTOS, your system is easily expanded by simply adding more tasks. Adding lower-priority tasks to your application will not affect the responsiveness of higher-priority tasks.

Safety-Critical Systems Certified

µC/OS-II has been certified by the Federal Aviation Administration (FAA) for use in commercial aircraft by meeting the demanding requirements of the RTCA DO-178B standard (Level A) for software used in avionics equipment. To meet the requirements for this standard, it must be possible to demonstrate through documentation and testing that the software is both robust and safe.

This is particularly important for an operating system, as it demonstrates that it has the proven quality to be usable in any application. Every feature, function, and line of code of µC/OS-II has been examined and tested to demonstrate its safety and robustness for safety-critical systems where human life is on the line.

µC/OS-II also follows most of the Motor Industry Software Reliability Association (MISRA) C Coding Standards. These standards were created by MISRA to improve the reliability and predictability of C programs in critical automotive systems.

Your application may not be a safety-critical application, but it's reassuring to know that the µC/OS-II has been subjected to the rigors of such environments.

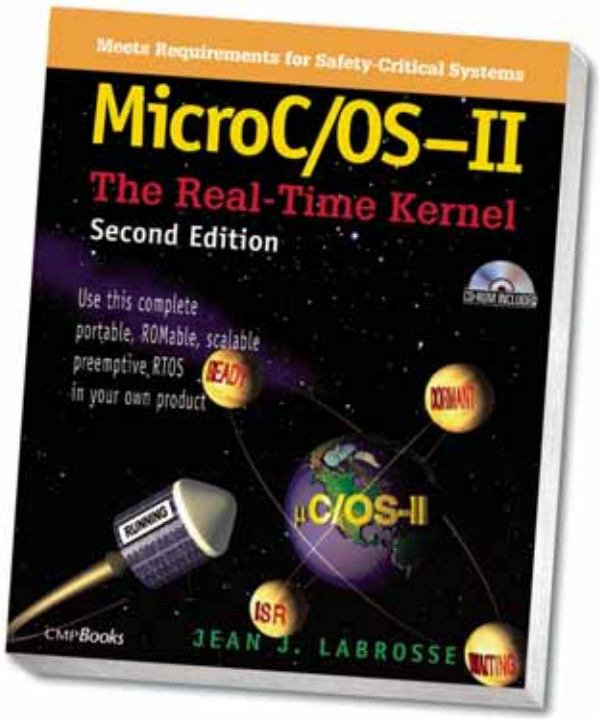


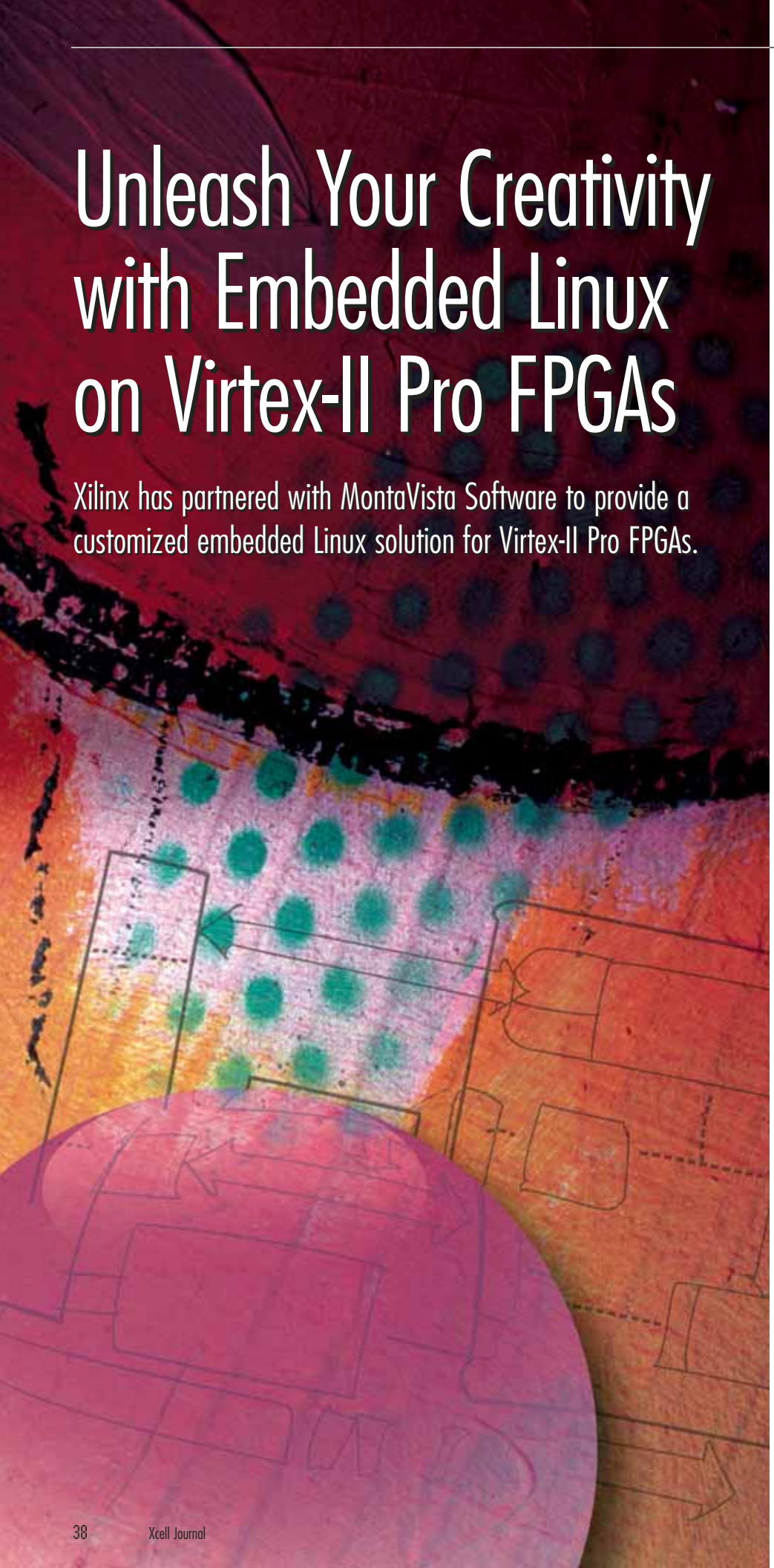
Figure 2 – MicroC/OS-II book cover

Conclusion

An RTOS is a very useful piece of software that provides multitasking capabilities to your application and ensures that the most important task that is ready to run executes on the CPU. You can actually experiment with µC/OS-II in your embedded product by obtaining a copy of the MicroC/OS-II book. You only need to purchase a license to use µC/OS-II for the production phase of your product.

Unleash Your Creativity with Embedded Linux on Virtex-II Pro FPGAs

Xilinx has partnered with MontaVista Software to provide a customized embedded Linux solution for Virtex-II Pro FPGAs.



by Milan Saini
Technical Marketing Manager
Xilinx, Inc.
milan.saini@xilinx.com

The open-source Linux® kernel with real-time extensions now provides most real-time operating system (RTOS) functionality and is rapidly emerging as a viable alternative to other commercial offerings. An RTOS is typically used to operate the majority of today's high-performance embedded systems.

The development of embedded applications on a single chip has reached new levels of integration, performance, flexibility, and ease of use with the introduction of Virtex-II Pro™ Platform FPGAs, which have up to four IBM™ PowerPC™ 405 32-bit RISC processors woven into the programmable fabric. The platform's sophisticated technology, however, typically requires newer tools and methodologies to address the unique characteristics of the device.

For those new to embedded Linux applications or Virtex-II Pro devices, this article explains some key concepts and answers commonly asked questions.

Key Linux Features

The open-source, royalty-free, and POSIX-compliant nature of Linux helps developers build highly customized and highly portable applications. Linux derives much of its value from gradually accumulating functionality through the combined contributions of its worldwide base of users. Similar to Unix in many ways, Linux offers a range of powerful operating system capabilities, including memory protection, processes, threads, and extensive networking facilities. It is generally considered to be a mature and robust technology.

The Linux kernel is licensed under the terms of the GPL (General Public License) with an important clarification: Code that employs the normal services of the Linux kernel via system calls or dynamic (module) linkage is considered normal use of the kernel and does not fall under the heading of "derived work." This means that dynamically linked

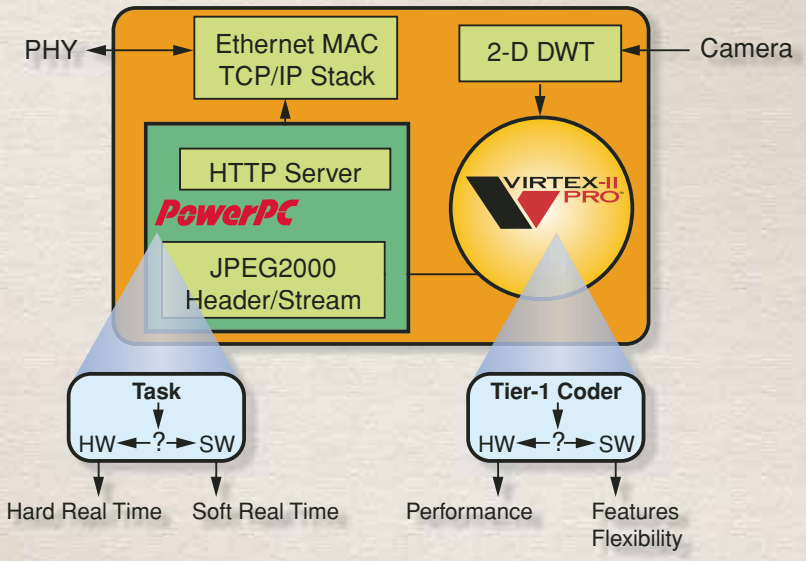


Figure 1 - Functions can be implemented on either programmable logic or PowerPC code.

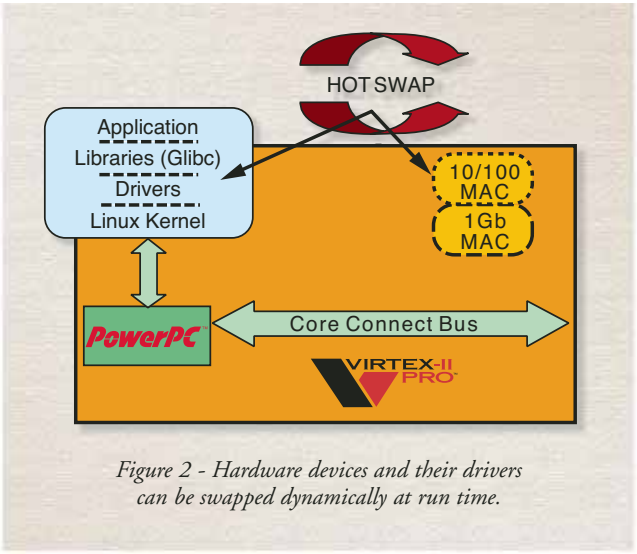


Figure 2 - Hardware devices and their drivers can be swapped dynamically at run time.

device drivers may not be covered by the GPL and their source code need not be disclosed to recipients of binary versions of the drivers.

In Linux space, "glibc," or the GNU C Library, is an important system call library used by most applications to provide an interface to the kernel. Glibc is distributed under the Lesser GPL (LGPL) license. Most applications link dynamically to this shared library to access the functionality of the kernel.

Linking to glibc is not considered to be linking directly to the Linux kernel – having glibc under LGPL frees applications linked to glibc from the GPL licensing requirements.

This flexible licensing model enables the mixing of proprietary drivers and application code with the Linux kernel and libraries, opening up a wider set of applications for use on Linux.

What Is Embedded Linux?

Embedded Linux distributions such as MontaVista® Linux® are usually configurable versions of standard Linux that make it more suitable for embedded systems development. Among the notable changes are the addition of a real-time scheduler and a preemptable kernel. These features make the interrupt and task response times shorter and more predictable. As a result, interrupt and task responses are more in line with the requirements of typical embedded applications. MontaVista Linux is currently the primary development environment for Virtex-II Pro-based embedded systems.

Strengths of Combining Linux and Virtex-II Pro

The configurable nature of Linux combined with the high versatility of Virtex-II Pro architecture offers designers unprecedented system flexibility and unique HW/SW co-design possibilities.

Because the platform is completely programmable, a design function can be implemented in either hardware (programmable logic) or software (PPC405) depending on the real-time design constraints and trade-offs between performance and flexibility, as shown in Figure 1.

Combining Linux and its reloadable modules with the partial reconfiguration feature of Virtex-II Pro FPGAs allows for features such as run-time device "hot-swapping," as shown in Figure 2.

For example, based on environmental factors, a design can swap a 10/100 Ethernet MAC with a faster 1 Gb MAC in the FPGA. Linux can reload the corresponding new device driver dynamically and at run time. This allows a design to adapt to its environment without the need for redesign and redeployment, which is a significant advantage.

Typical Design Flow

Development of embedded applications on Virtex-II Pro devices typically involves the following:

- Assembling a hardware platform
- Configuring the Linux kernel
- Developing the firmware (device drivers) and software application code
- Downloading the design into the FPGA device or board.

The Xilinx Embedded Development Toolkit (EDK) and integrated software environment (ISE) software help a hardware engineer assemble a reference hardware platform. This process involves the instantiation, initialization, and configuration of the processor, bus, and peripheral components. The EDK tools (platgen, libgen) write both the hardware implementation netlists and the software header file. The header file named "xparameters.h" captures and stores customized information such as device ID, base address, and so forth for all peripherals in a single location. The information contained in this file is referenced by the kernel and drivers and is essential to building the Linux application.

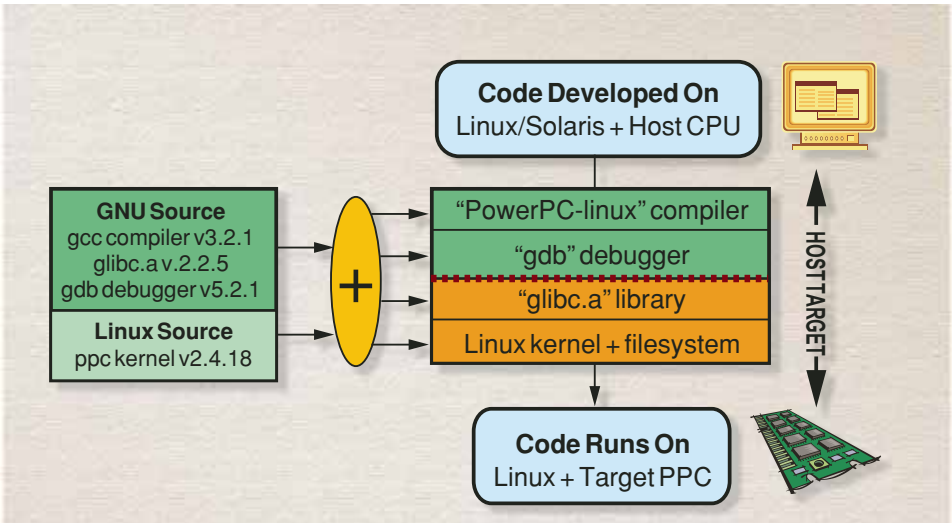


Figure 3 - The Linux development environment

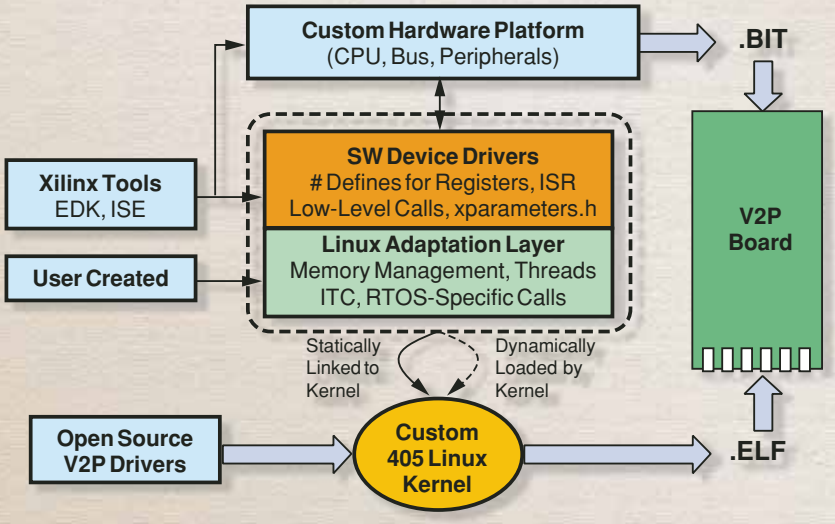


Figure 4 - Linux device drivers can be built on top of lower-level drivers provided by Xilinx.

Using one of several Linux configuration tools (such as the MontaVista Target Configuration Tool [TCT]™ GUI-based utility), the Linux kernel is configured by making appropriate selections of drivers and services required by the application software. In general, the idea is to keep the kernel image size as small as possible by excluding features not needed by the application. (Linux kernel configurations typically require between 2 and 8 Mb of memory.)

Application software for Linux can be developed and debugged with GNU C and C++ compiler tool chains to create binary "elf" images that can be deployed onto a Virtex-II Pro board.

Various methods for debugging applications and downloading into the FPGA are discussed later in this article.

Embedded Linux Environment for Virtex-II Pro

Xilinx has partnered with embedded Linux pioneer MontaVista Software to provide a customized embedded Linux solution for Virtex-II Pro devices. Users receive the embedded Linux kernel, Virtex-II Pro device drivers, host and target tools, applications, utilities, and product support. Support includes updates, enhancements, defect corrections, training, and technical assistance. Some of the development tools are shown diagrammatically in Figure 3 and include:

- Cross-compiler tool chain. To increase efficiency, developers use cross-development tools that run on faster and friendlier host machines. The code is compiled to run on the actual target CPU. MontaVista provides these tools for multiple Linux and Solaris hosts.
- Libraries. Glibc and glibc-linux threads are the most commonly used set of libraries for Linux application development. These libraries contain the core set of functions to which most utility and application code links.
- Kernel and drivers. The kernel provides core functionality such as memory management, threads, and I/O behavior. A "makefile" included in the distribution package is used to assemble the Linux kernel per the user and application requirements.
- The target Linux file system. A kernel without programs to run is not useful, so a set of programs and utilities have been developed to run on Linux for the target PPC405 CPU. Examples of such programs include bash, tcsh, perl, awk, sed, grep, make, gcc compilers, a vi clone, emacs, and X Windows.

The above tools and utilities are packaged and distributed by MontaVista in its Professional Edition. MontaVista also provides a Preview Kit, a free evaluation version of this product.

The Linux kernel, libraries, and tools can also be compiled from the open source, but this configuration is not supported by Xilinx.

Prototyping Boards and Design Availability

Xilinx and its partners have made several Virtex-II Pro evaluation boards available, including the ML300 (V2P7) from Xilinx and the Insight V2P4/7 from Insight Electronics. The boards differ mostly in external-to-FPGA component configuration, with the ML300 offering, in general, a richer set of processor companion devices. These boards can be used for evaluation, demonstration, or development. Full details can be found by following the links at the end of this article.

The MontaVista Linux kernel and drivers

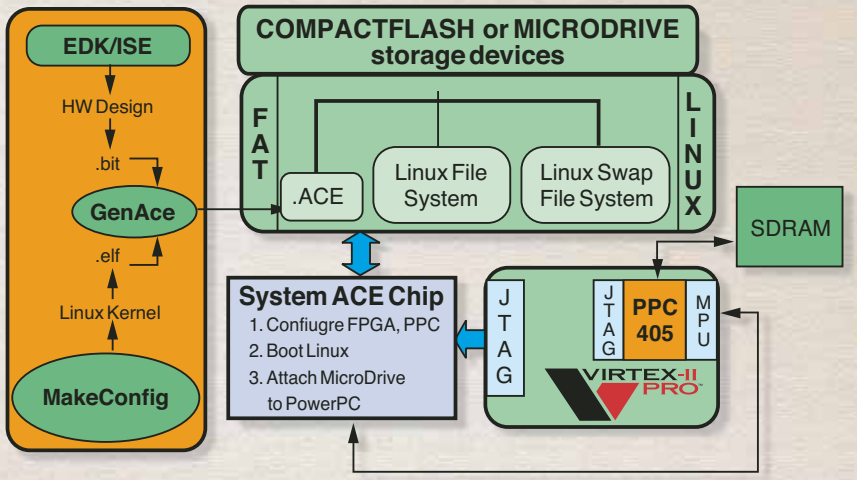


Figure 5 - The System ACE companion chip provides a powerful and easy-to-use method to boot both the processor and the FPGA.

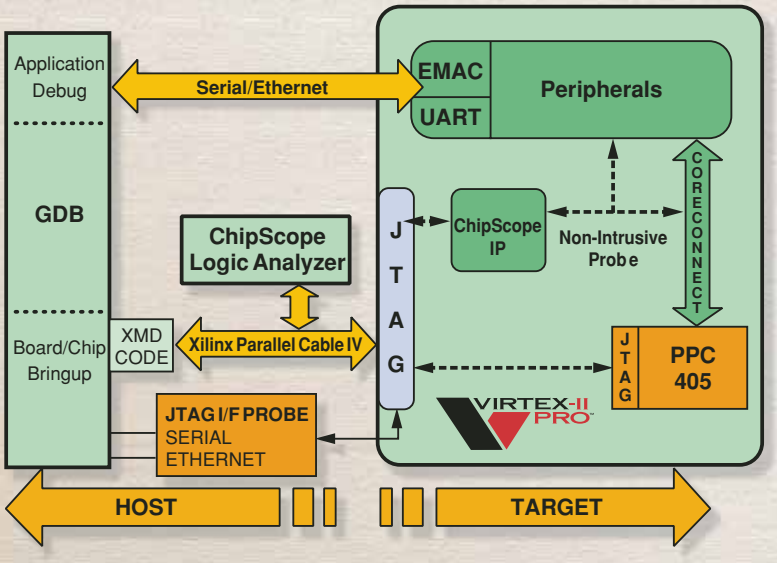


Figure 6 - GNU gdb can be used in various connection schemes in combination with the ChipScope Pro Logic Analyzer tool to offer full debug visibility.

have been tested with the ML300 Reference Design using EDK (Design Source: *http://www.xilinx.com/ise/embedded/edk_examples.htm*). This development platform provides a reasonable starting point for designing custom applications. The same design should run on other Virtex-II Pro boards without modifications.

Linux/Virtex-II Pro Device Drivers

A device driver is code that provides an abstraction layer for hardware I/O devices so that higher levels of software can access the devices in a uniform and hardware-independent fashion. Since Virtex-II Pro FPGAs allow processor peripherals to be uniquely customized, the corresponding device-driver code must match the user-selectable configuration of the device.

The custom and dynamic nature of the embedded platform assembly is an important difference between Virtex-II Pro development and traditional embedded development on boards and processors with static peripheral configurations. To speed development, Xilinx EDK tools have been designed to create custom software device drivers for all relevant hardware peripheral IP blocks Xilinx provides. The driver code is modular and generic and is designed to be easily adapted to higher-level OS functions, as shown in Figure 4.

Several of the Xilinx-provided device drivers have been adapted for Linux and released as open source. Among them are: 16450/16550 UARTs, 10/100 Ethernet, PS/2 for mouse and keyboard, the System ACE™ CF configuration storage device, GPIO, PCI, TFT display, and touchscreen. The drivers can either be built directly into the kernel or dynamically linked as loadable modules.

Loading the Application into the Target FPGA

A typical boot sequence after power up is as follows:

1. Load FPGA bitstream
2. PPC boot
3. Load Linux kernel from Flash memory
4. Start application.

While an application is in active development, a software debugger like GDB can be used to load the application to the target CPU through the JTAG pins of the processor.

On production boards, several methods can be used:

- Recommended method – use the Xilinx System ACE CF companion chip that can read the combined FPGA bitstream and Linux kernel image from a CompactFlash™ mass storage device or MicroDrive™ compact storage device. The System ACE CF chip configures the FPGA, boots the CPU, and then loads the kernel image into processor memory, as shown in Figure 5. The PowerPC 405 program counter is advanced to the starting address of the Linux kernel. After the system is up and running, the processor can use the CompactFlash/MicroDrive as its secondary storage device. Other benefits of this process include the flexibility to store multiple boot configurations on the FAT file system and eliminating the need for memory at the reset vector.
- PPCBoot™/bootloader – a bootloader is a program that runs just before the operating system really

starts to work; in fact, it loads the kernel of the operating system. PPCBoot is a general-purpose bootloader for embedded PowerPC boards, supporting more than 80, including boards based on Xilinx Virtex-II Pro FPGAs. It offers many features, including hardware initialization, loading kernel image, trivial file transfer protocol (TFTP) download, flash write access, and low-level diagnostics. PPCBoot is available from *ppcboot.sourceforge.net*.

Debugging Tools

For initializing the board, a host-based debugger such as GNU gdb with its graphical front end ddd (or products from emacs, Insight, and others) can non-intrusively control the PPC405 CPU. The processor's JTAG pins are connected through a debugging interface such as Abatron's BDI2000 or Lauterbach's Trace32™ debugger, as shown in Figure 6. Connection can also be made through a Xilinx-provided resident host program called "xmd" in series with the Xilinx Parallel Cable IV. GDB has full visibility and control of the processor memory, caches, and TLB entries.

Using a Parallel IV cable, the ChipScope™ Pro Logic Analyzer tool from Xilinx helps provide visibility and debug features for corresponding hardware logic where most of the processor peripherals reside. Using a single cable to control and observe both the hardware and software not only simplifies the lab setup but also provides a comprehensive, powerful, and efficient system-debug methodology.

Conclusion

For data sheets, application briefs, and additional information on the ML 300 evaluation board, visit Xilinx online at *www.xilinx.com/ml300*, or for information on Virtex-II Pro FPGAs, visit *www.xilinx.com/virtex2pro*. For similar information on the Insight V2B evaluation board, visit *www.insight-electronics.com/virtex2pro*.

Additional information on MontaVista's Professional Edition 3.0 can be obtained by visiting MontaVista online at *www.mvista.com*.

MontaVista Linux and Virtex-II Pro Speed Your Design Cycles

by Bill Weinberg
Director of Strategic Marketing, MonteVista
bill@mvista.com

Developers are drawn to embedded Linux for a variety of reasons, including its:

- Familiar and robust UNIX/POSIX programming model and APIs
- Flexibility to modify source code
- Excellent throughput and "gold standard" networking
- Universality – no "lock-in" to a single proprietary supplier
- Reduced development and deployment costs from its open source base.

In several key aspects, embedded Linux is a lot like an FPGA device. Both are flexible, configurable, and cost-effective; both integrate a broad array of functionality and programmability into a single system; and both give you a jump start on system development and deployment.

Just as Xilinx provides high-performance, off-the-shelf CPU cores in a programmable matrix of logic and peripherals, MontaVista Software offers an off-the-shelf, ready-to-deploy embedded Linux platform for the Virtex-II Pro™ platform. Together, they decrease your time to market and slash development costs.

MontaVista Linux Professional Edition

MontaVista® Linux® Professional Edition delivers a development environment and Virtex-II Pro deployment platform.

Complete with IDE, cross compilers, debuggers, and configuration tools, the MontaVista Linux development platform offers real-time responsiveness, a scalable footprint, rich networking, hundreds of utilities, and thousands of deployable software capabilities. A MontaVista Linux product subscription lets your team focus on your application, not on reinventing Linux.

Features

MontaVista Linux Professional Edition targets the Virtex-II Pro FPGA with the following features:

- A kernel optimized for the IBM™ PowerPC™ 405GP processor, with full memory management unit (MMU) support and drivers for Virtex-II Pro physical and virtual peripherals
- Comprehensive board support for the Xilinx ML300 (V2P7)
- Easy transition to your application-specific board-level hardware
- Support for dynamic loading and field upgrade/reconfiguration of system software and device drivers for Virtex-II Pro platform reprogrammable peripherals
- Real-time application response with a fully preemptable kernel, a fixed-priority low-overhead scheduler, yielding low interrupt, and scheduling latencies
- Support for kernel and driver debugging with popular JTAG run-control devices
- Optimizing C and C++ cross compilers, debuggers, and other tools for the embedded PowerPC 405GP cores in the Virtex-II Pro family
- Cross development from Linux, Solaris, and Windows workstations.

With the MontaVista Linux development platform, your out-of-the-box experience is a snap – the MontaVista Linux Professional Edition tool kit installs easily on your development host and lets you boot your Virtex-II Pro target system with Ethernet-based netboot (TFTP), over a serial link, or with a JTAG bridge.

Within minutes, your development hardware appears on your local network as a peer system, speeding cross compilation and execution by mounting your development file systems (NFS or Samba) or by exporting its own RAMdisk software driver (over FTP, NFS, or Samba).

Once your team is ready to "cut the cord," the MontaVista Linux development platform lets you build a fully customized Linux kernel and optimize the footprint of your application and run-time libraries, ready to cross-deploy to your choice of disk, flash, network, or RAM-based file systems.

Conclusion

Whether you're building communications systems, control applications, or intelligent client devices, the integration and programmability of Virtex-II Pro Platform FPGAs with MontaVista Linux Professional Edition will streamline your development effort.

The same features also reduce your bill of materials costs, translating into faster time to market and profitability. The peer-level field-programmable nature of Linux-based Virtex-II Pro platforms also extends the life of your embedded applications, while lowering cost of ownership.

To learn more about MontaVista Linux Professional Edition for Virtex-II Pro, visit *www.mvista.com*, contact MontaVista Software headquarters at (408) 328-9200, or email *sales@montevista.com*.

Thousands of Designers. Infinite Applications. *One Powerful Resource.*

Powered by eSP WEB PORTAL

SPARTAN-3 FPGA

Online innovation

Whether you're designing next-generation consumer, digital video, automotive, industrial, or even broadband applications, the eSP web portal can help make your innovation a reality. Featuring everything from technology tutorials, market overviews, and system block diagrams to reference boards, IP cores and more, this is the ultimate resource for all your design solutions.

The power of 3

The high-performance, low-cost Spartan-3 platform FPGA features a range of density options from 50,000 to 5 million gates, as well as the lowest I/O and gate costs available. This makes it the ideal device for a wide variety of applications. Now, our eSP web portal gives you access to a complete suite of Spartan-3 silicon, software, and IP solutions optimized for your design.

For more information, visit us at ***www.xilinx.com/esp/s3.***

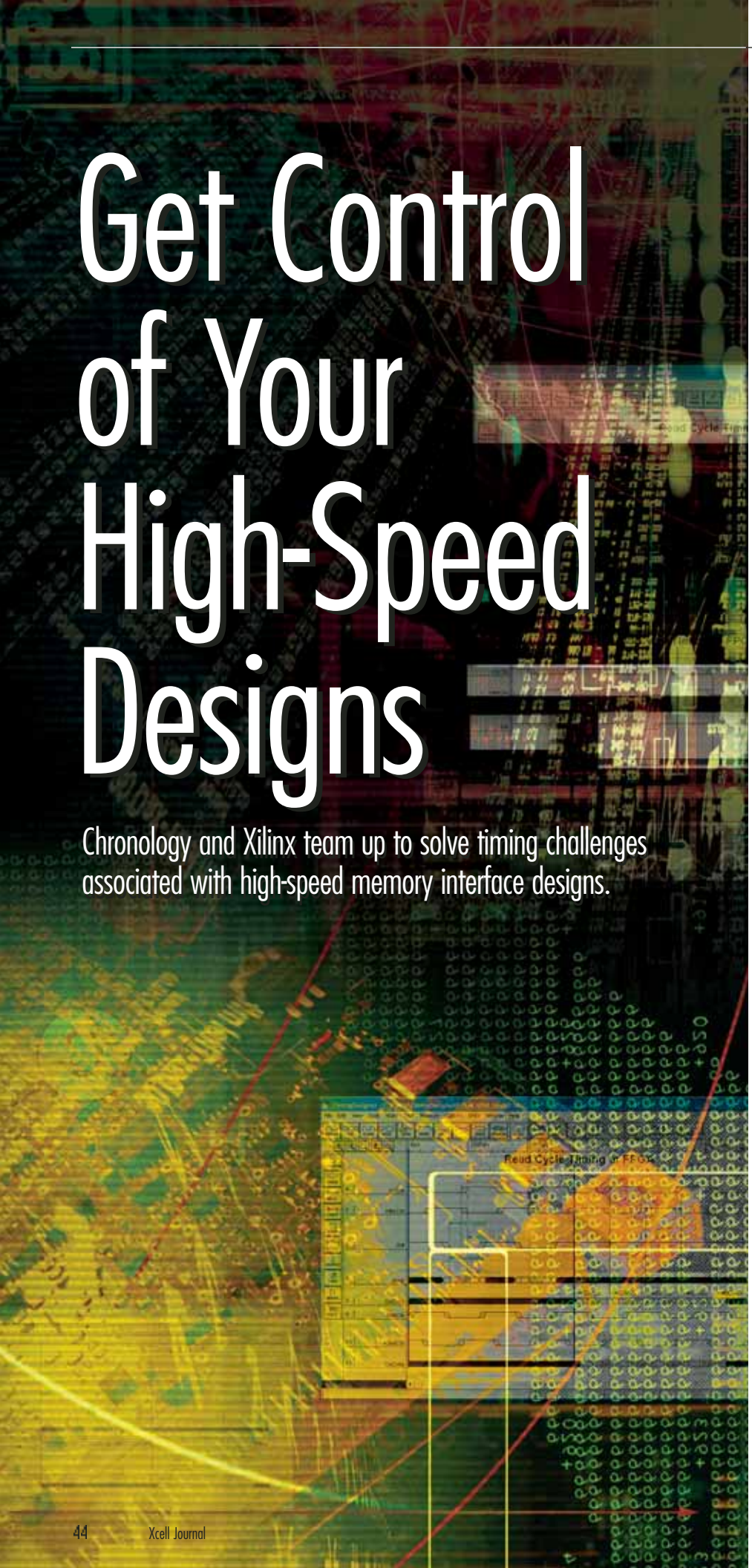
XILINX®
The Programmable Logic Company℠

www.xilinx.com/esp/s3

FORTUNE® 2003
100 BEST COMPANIES TO WORK FOR

Get Control of Your High-Speed Designs

Chronology and Xilinx team up to solve timing challenges associated with high-speed memory interface designs.



by Jerry A. Long
Technical Marketing Manager
Chronology Division, Forte Design Systems
jlong@forteds.com

It's a given – the ever-increasing demand to get your designs to process more information in less time. Designing with new high-speed technology helps you meet market demands, but it's likely you're also encountering difficult interface design challenges. Timing issues previously deemed insignificant are now impacting design schedules and can no longer be overlooked.

Design teams are seeking extremely efficient, low-latency components that can operate at very high frequencies. With the introduction of QDR™ (quad data rate) SRAMs (co-defined and developed by members of the QDR SRAM Consortium and Xilinx), designers now have first-rate performance devices available for their design projects.

Created for systems operating above 200 MHz on Xilinx Virtex™-II FPGAs, QDR SRAM devices provide an unprecedented four data transactions per clock cycle. QDR SRAM devices implemented on Virtex-II Pro™ Platform FPGAs include advanced features that support the design of high-speed memory interfaces.

To fully take advantage of the high-performance features of these new devices, however, you must account for the intricate timing issues associated with high-speed interface design (Figure 1), including:

- Analyzing timing options
- Incorporating signal integrity and physical effects into timing considerations
- Managing and monitoring timing margins throughout the design process.

If you isolate timing considerations within the confines of each stage of a project, the design can be extremely costly in terms of time, effort, and resources. You simply cannot compromise accurate exchange of timing-critical data among engineering teams without putting design specifications and release schedules at risk.

Chronology's interactive TimingDesigner™ specification and analysis tool meets the stringent demands of designing high-speed memory interfaces implemented on Virtex-II and Virtex-II Pro Platform FPGAs.

New Design Challenges

Along with the increased throughput advantage of high-performance devices comes the inevitable interface design challenge of ensuring accurate data transfer.

QDR SRAM devices allow four data transactions per clock cycle by providing separate read and write data buses, each with DDR (double data rate) performance characteristics. Separate data buses require:

- Twice the I/O pin requirement for data
- Two clocks
- "Center-aligned" data presentation.

Virtex-II Pro technology allows you to leverage these features with its abundant I/O resources and high-performance clock management circuitry. However, the benefit of designing with a Virtex-II Pro device does not by itself address all the timing challenges.

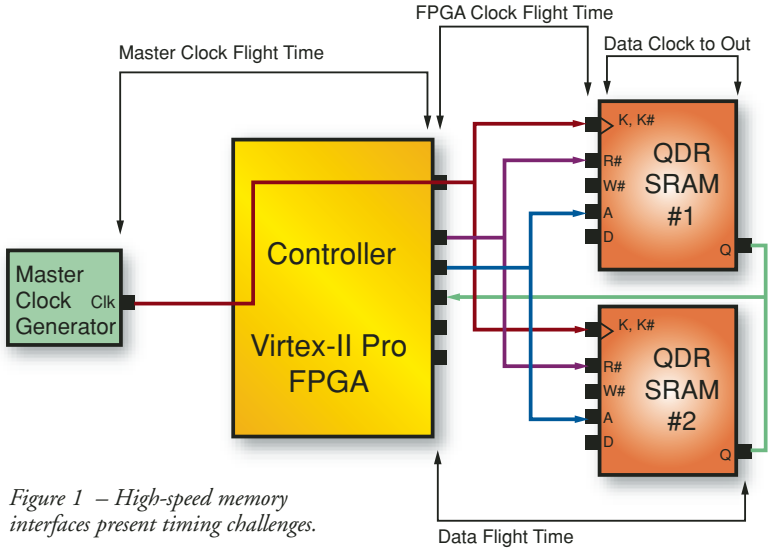


Figure 1 – High-speed memory interfaces present timing challenges.

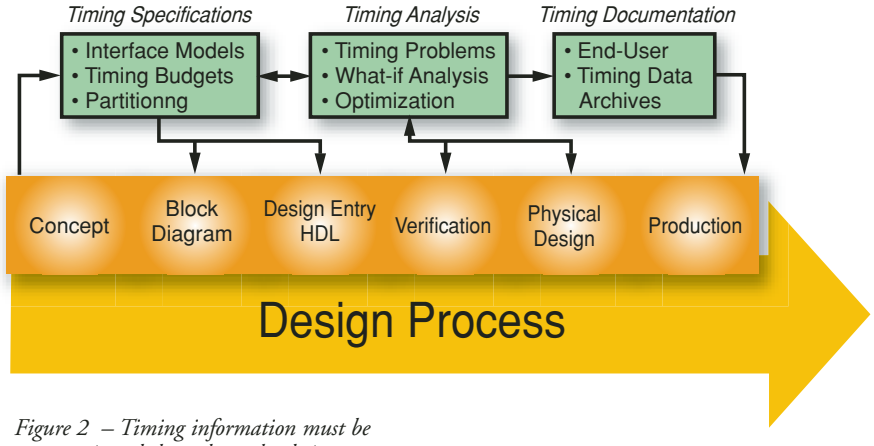


Figure 2 – Timing information must be communicated throughout the design process.

Ever-Shrinking Timing Margin

As operating frequencies rise beyond 200 MHz, the period within which data can be captured and presented decreases to 5 ns or less. At these frequencies, the margin for setup and hold times dwindles. Shrinking margins mean less room for securing an accurate data capture and presentation window.

Further complicating matters are increased crosstalk susceptibility and transmission line effects that result from higher operating frequencies. Signal integrity issues are a significant factor in high-speed designs, and you must monitor them as your design progresses. The combined effects of parasitic PCB traces, IC packaging, bond wires, and physical die characteristics adversely impact signal quality. These effects require additional settling time, therefore shrinking the timing margin you have available for reliable data capture.

Communication Is Critical

Project teams cross multiple functional organizations, each with their own sets of software tools and workflows to meet their specific design needs. For high-speed design, it is imperative to monitor and manage timing margin limitations throughout the design process. This places new automation and accuracy requirements on the exchange of design specifications among engineering teams.

Too often, critical timing margin information is miscommunicated among engineering teams. These teams depend on text documents created by manual interpretation of large complex report files that were generated by tools designed for other uses. Such timing documents are often laborious to maintain.

They also introduce the opportunity for error, especially because timing margin information changes frequently throughout the course of a design project. To reduce the risk of error, teams must have a common, automated means to accurately communicate and exchange timing data (Figure 2).

TimingDesigner Analysis Tool

When designing high-speed interfaces, you need:

- A means to detect timing problems early
- The ability to display visual representations of design requirements
- A tool to easily implement alternative solutions.

The TimingDesigner interactive timing specification and analysis tool is being used by interdisciplinary project teams when designing high-speed interfaces for Virtex-II and Virtex-II Pro FPGAs and QDR SRAM memory devices. TimingDesigner allows you to create interactive timing diagrams for capturing interface specifications,

analyzing component interface timing, and communicating design requirements among project engineering teams.

Timing Diagrams and Issues

The way TimingDesigner incorporates the use of timing diagrams is innovative and extremely versatile. You can create diagrams that focus on interface protocol specifications, where each functional protocol operation is represented with a single diagram.

Alternatively, you can create diagrams to focus strictly on a particular signal path, with each signal representing a different propagation point along the entire critical path. TimingDesigner's built-in static timing analysis engine immediately calculates and reports worst-case timing margins as you explore design alternatives. This allows you to easily model, analyze, and manipulate the most critical parts of your design.

Component Diagrams

When you want to examine high-speed memory interface timing based on your design components, TimingDesigner generates a separate diagram for each component involved. Each diagram represents the particular operation of the interface protocol, such as a read operation for the QDR SRAM.

Additionally, each diagram includes related signals and the timing requirements associated with the operation. Event relationships such as delays for control signal events, data signal setup and hold requirements, and clock jitter tolerances are also represented in the TimingDesigner diagram.

This feature gives you quick and accurate visual reference for the interface requirements. It also allows you to create a complete interface protocol specification while promoting diagram re-use as multiple diagrams accumulate in an interface protocol library.

Once these component-based diagrams are drawn, you can easily merge them to perform timing analysis for the entire interface circuitry (Figure 3). The receiving component interface diagram is linked to the initiating component diagram via delays associated with anticipated PCB trace information. You can also create additional signals and clocks at incremental points along the timing path. This technique provides an added degree of visibility of signal propagation delays, helping to illustrate complete timing closure for critical timing paths.

Signal Path Diagrams

When you need to focus on individual critical paths, TimingDesigner can create timing diagrams based on designated signal paths. In this case, you create a signal for each point of interest along a specific signal propagation path. By doing this, each stage that may influence the designated signal – such as I/O buffer delays, PCB flight path delays, signal integrity delays, or other known influences – is illustrated in the critical path analysis. This approach reveals the impact on timing margins, and ultimately, the source of timing failures. The capability to trace each critical path from start to finish gives you confidence in achieving timing closure.

Other Timing Diagrams

You can also use TimingDesigner diagrams to accurately represent complicated clock phase and de-skew relationships, which are required for the design of QDR SRAM clocking signals.

Using the various clock features provided in TimingDesigner, such as accurate jitter representation and derived clock elements, you can generate diagrams that can illustrate the clocking characteristics needed for practically any high-speed interface. You can then easily recognize the clocking requirements and confidently program the digital clock managers (DCMs) in Virtex Platform FPGAs for dependable QDR clocking schemes.

Built-In Static Timing Analysis

No matter what style of timing diagram you choose to implement, TimingDesigner's built-in static timing engine delivers accurate signal timing analysis at any stage in the design process. At each design stage, the static timing analysis engine can instantly trace all of the delay paths specified in the timing diagrams, remove common uncertainties, adjust for track delays, select critical paths, compute worst-case timing margins, and flag violations.

This intuitive use of timing diagrams offers quick evaluation of design alternatives – in addition to fast calculation of timing margins and insight into solving complex timing problems early in the design process.

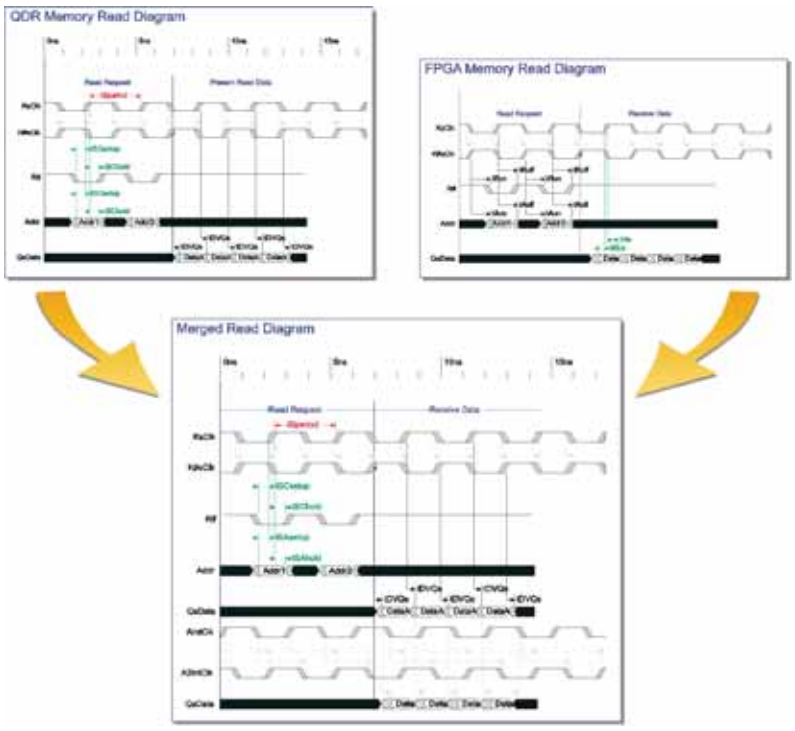


Figure 3 – Merged diagrams allow for complete interface timing analysis.

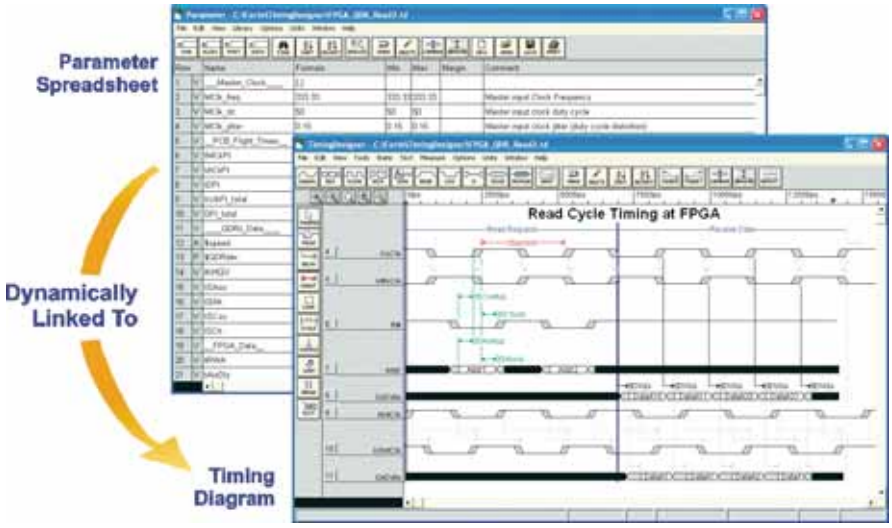


Figure 4 – A dynamically linked spreadsheet provides immediate feedback for easy exploration of timing alternatives.

What-If Timing Analysis

All edge relationship events entered in a TimingDesigner diagram (for example, delays, constraints, guarantees, and measures) are automatically generated in a dynamically linked Parameter Spreadsheet (Figure 4). The tabular format of the timing events represented in the diagram enables you to easily manipulate value fields, as well as comment fields, for event descriptions.

The Parameter Spreadsheet also supports the parameterization of timing diagrams with variables to represent various timing scenarios. These variables can be referenced from events in the diagram.

For example, clock characteristics such as frequency, period, phase shift, and jitter can be represented with variables in the spreadsheet, and then used in other formulas to create cycle-accurate timing relationships. This aspect of TimingDesigner provides added flexibility when designing QDR SRAM interfaces that require center-aligned data or when illustrating various clocking signals of your Virtex-based design.

The Parameter Spreadsheet gives you the ability to create complex formulas to model path delay variables, signal loading, temperature, and other elements that may impact timing relationships.

Using variables in the Parameter Spreadsheet facilitates quick and accurate "what-if" analysis. Throughout the design process, you can evaluate circuit functionality and performance using different sets of parameterized values, providing a highly flexible and productive design environment.

For example, you can evaluate numerous operating frequencies by simply changing the variable used for clock frequency. You can also use this method to evaluate multiple Virtex-II and Virtex-II Pro clocking schemes by providing variables for phase relationships to determine the most appropriate phase increment for the DCMs.

Library Spreadsheet Flexibility

To further expand timing analysis flexibility, you can use the TimingDesigner Library Spreadsheet to create part-specific timing information that can be used by any timing diagram. This is especially useful for devices where a single diagram (or set of diagrams) can model the device functionality and then reference a Library Spreadsheet for specific speed grades or voltage grades.

Expanding on this concept, you can easily switch in and out of libraries of different timing relationships to evaluate cost and performance trade-offs and help determine the best speed grade option for a specific design.

Timing libraries are also very effective when analyzing pre-route versus post-route timing characteristics of a design. You can use post-route timing reports from Xilinx ISE tools (or other layout tools) to create a TimingDesigner library. Because the library file format is a comma-separated ASCII file, creating multiple timing libraries specific to a particular routing run is easy. You can insert these libraries into the existing timing diagram analysis for a post-route confirmation that critical timing paths have been satisfied.

TimingDesigner Saves Time

TimingDesigner diagrams are an excellent documentation medium for interface specifications. The easy-to-understand format of these diagrams reduces the chances for error due to misinterpretation by other project team members. The format provides a clear and concise view of available timing margins.

Each design team can interact with the timing diagram, making available their incremental contribution to the overall timing path result. Thus, you get immediate notification of potential timing problems that may be uncovered during different stages of your design process.

Using TimingDesigner's OLE capability, timing diagrams can be used in their native format or embedded or linked into commonly used publishing and presentation tools. This provides a simple way to convey critical timing path information among design groups. Additionally, TimingDesigner exports common image formats, including MIF, EMF, TIF, and EPS. Import of VCD and FSDB simulation files is also supported.

Conclusion

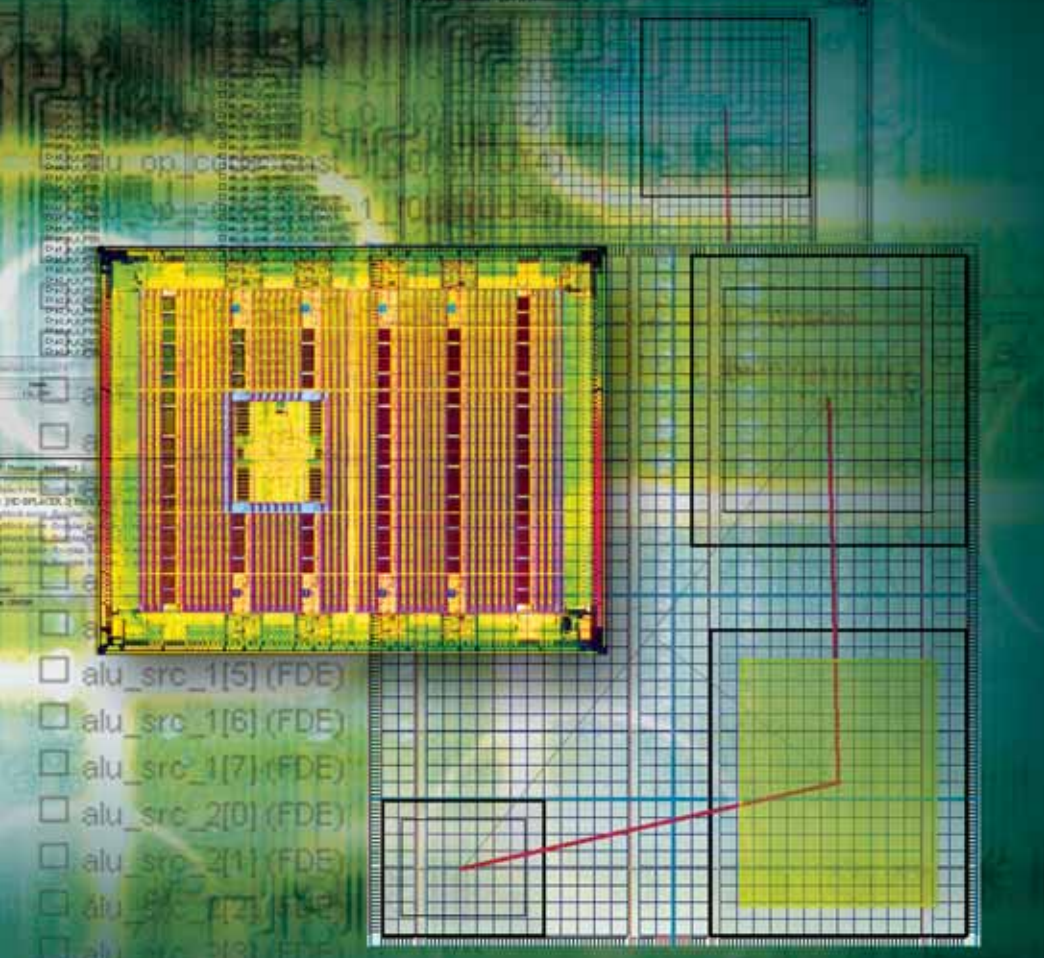
Together, the Chronology Division of Forte Design Systems and Xilinx provide a powerful solution set that allows you to tackle the timing challenges of high-performance design. High-speed designs often have stringent specifications and tight release schedules, so you need an interactive timing specification and analysis tool to obtain fast and complete timing margin analyses – addressing the factors that can ultimately affect your design success.

The versatile clock configurations and abundant I/O resources of the Virtex-II Pro Platform FPGA allow high throughput data transfer, while TimingDesigner delivers accurate critical path timing analysis results required for interface design with high-speed memory devices such as the QDR SRAM.

To learn more about TimingDesigner and the Chronology Division of Forte Design Systems, please visit *www.timingdesigner.com.*

Use ASIC Design Methodology for Your Next FPGA Design

Hier Design's PlanAhead software eliminates many problems encountered in high-end FPGA designs.



by Dino Caporossi
Vice President, Marketing
Hier Design, Inc.
dino@hierdesign.com

As an FPGA designer, you probably face many of the same problems that have plagued ASIC designers, including lengthy, repetitive place-and-route (PAR) runs; unpredictable route times; and difficulty maintaining performance throughout frequent design iterations. These problems can delay or even prevent you from completing your design, resulting in increased engineering costs and longer time to market.

ASIC-style methodologies, however, enable you to divide and conquer your complex FPGA designs by breaking them into component blocks, shortening the length and increasing the predictability of routing. As shown in Figure 1, these methodologies also allow you to analyze, detect, and correct potential implementation problems before PAR.

An ASIC-style methodology offers the following benefits:

- Quicker incremental design changes or engineering change orders (ECOs)
- Faster, more predictable PAR
- Fewer iterations
- Improved performance
- Tighter utilization control
- Better teamwork
- IP reuse.

To give you the advantages of ASIC design techniques, Hier Design Inc. has developed the PlanAhead™ hierarchical design and physical optimization tool suite to maximize your FPGA design efficiency.

Quicker Incremental Design Changes

When you make even minor changes to a given logic block with a flat methodology, you must redo place-and-route for the entire design. With today's larger FPGA netlists, 50 or more PAR iterations – at eight or more hours apiece – are common. This adds up to a significant amount of wasted time. With an ASIC-style design

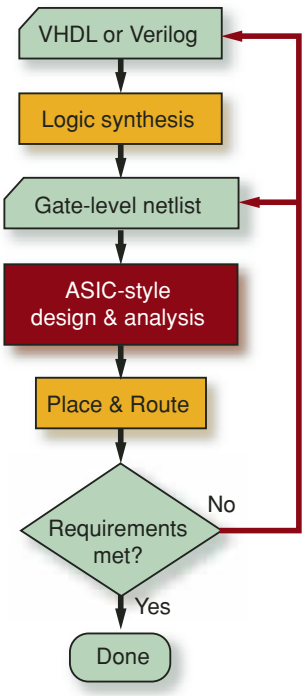


Figure 1 – An ASIC-style design methodology reduces the number and length of FPGA design iterations (highlighted in red).

methodology, you can use hierarchy to reduce PAR time. By breaking your designs into smaller pieces, or blocks, you don't need to run PAR on the entire design each time you make an incremental design change. Instead, you can just run PAR on the block or blocks that have changed, leaving the rest intact.

As an example, see Figure 2. In the flattened methodology on the left, a logic block is highlighted in yellow. Notice how it is scattered over nearly the whole chip. Any design change in this logic block means that you must PAR the entire flattened design all over again. Consequently, design performance may change significantly.

In the hierarchical methodology on the right, the same logic block is highlighted in yellow. Note that it is localized within a distinct area of the chip. If you make a change to this logic block, you only need to PAR this block. The rest of the design and its performance remain largely unaffected.

Faster, More Predictable PAR

Without hierarchy, PAR algorithms on a flat design are less efficient and need far more time to operate. Yet traditional FPGA design tools still primarily operate in that mode.

An ASIC-style methodology allows you to define area groups. These groups break the design into smaller pieces that PAR algorithms can more easily handle. Because individual blocks are more manageable, PAR takes less time to complete the overall design.

In the traditional FPGA design process, the duration and number of PAR runs have not only become a bottleneck – the results themselves are unpredictable and unreliable. Even if you make no design changes, one run may produce a faster or slower design than previous runs because of the randomness of the routing algorithms.

Some designers using flat methodologies try to reach timing goals using multiple routing runs, each with different random seed values, hoping that one of them will produce a design with adequate performance.

A hierarchical methodology, however, offers a more deterministic process by enabling you to define area groups to steer PAR toward acceptable timing.

You can also lock in placement results for individual blocks that have already met timing goals, so that subsequent PAR iterations do not change the performance of the locked blocks. This serves to further stabilize the PAR process, ensuring more reliable and predictable results.

Fewer Design Iterations

If you are designing large FPGAs, you probably iterate the physical implementation many times to meet multiple simultaneous requirements. Such requirements may include:

- Connectivity
- Utilization
- I/O placement
- Clock regions
- Power
- Timing.

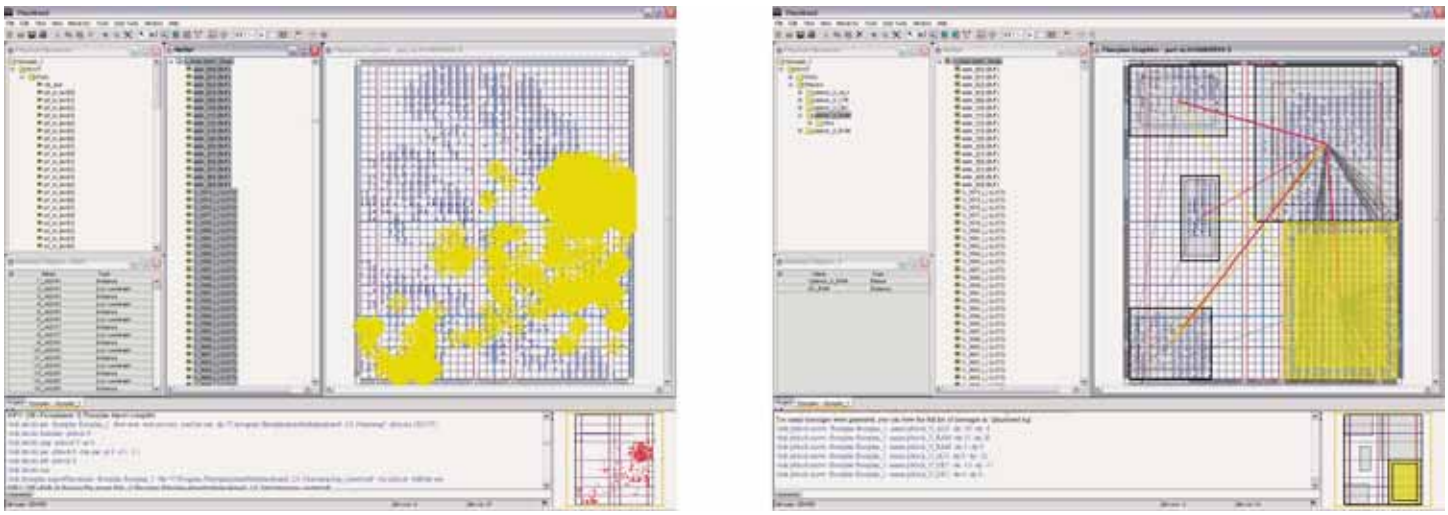


Figure 2 – A flat design methodology versus an ASIC-style, hierarchical methodology

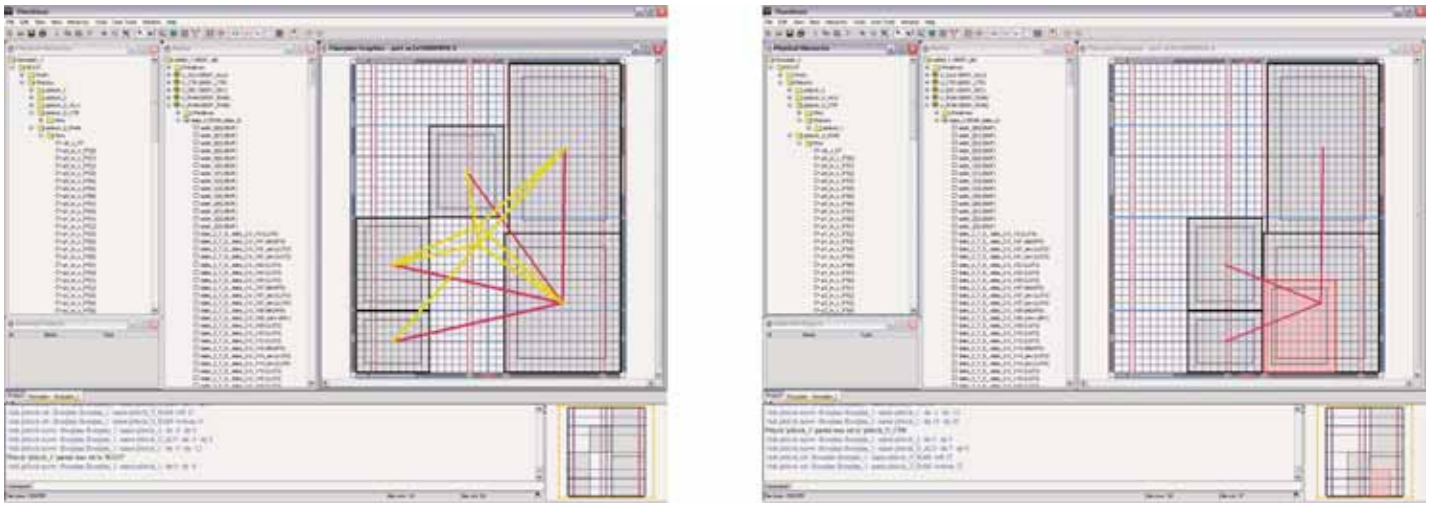


Figure 3 – Avoiding routing congestion through early analysis

Without an ASIC-style methodology, there is no way to know whether all these requirements can be met until after PAR. What's worse, many of the requirements are interrelated, so making design changes to achieve one requirement often prevents you from achieving several others. Consequently, you spend too much time iterating your designs.

ASIC-style methodologies help you quickly reach your requirements – prior to PAR – through early design analysis, tightly coupled with floorplanning. With floorplanning and early analysis, you can substantially reduce the number and length of PAR iterations.

In Figure 3, early connectivity analysis in the PlanAhead software (left screen) provides visual clues: The thickness and color of bundled lines represent the nets between blocks. You can see – and fix – routing congestion before PAR. You can quickly adjust the floorplan by moving, rearranging, and combining blocks in the physical hierarchy (right screen). Moreover, you can manipulate the physical hierarchy independently from the logical hierarchy.

Using this process, the PAR algorithms now have the necessary partitioning, block arrangement, and constraints to guide them to success.

Improved Design Performance

When designing with today's large FPGAs, you may encounter timing knots, long critical paths spanning hierarchy, complex clocking, and high fan-out – making timing goals more difficult to meet. ASIC designers have successfully overcome these kinds of problems through techniques now available to FPGA designers.

Static timing analysis enables you to quickly visualize potential timing problems early in the design cycle, before PAR. Floorplanning lets you make adjustments to avoid these timing problems by controlling logic migration and shortening connectivity paths.

Static timing analysis also highlights critical paths in the floor plan so that you can fix them by using constraints or manually rearranging the path instances.

In a misguided effort, some ASIC-style methodologies attempt to improve performance by floorplanning before synthesis. This is not a good idea, because there may be large errors in the estimates of design performance. In contrast, PlanAhead software acquires more accurate timing information at the highest impact point in the design flow, thereby maximizing performance improvement.

The PlanAhead software also includes physical optimization capability to address implementation problems that may arise later in the design cycle, post-PAR. With the PlanAhead tools, you can still adjust placement at the block or leaf level to reach performance goals. You can then run timing analysis to see if performance has improved. With this process you get nearly instantaneous feedback, rather than waiting for hours for PAR to finish.

Methodology Results Example

Table 1 demonstrates the advantages of adopting an ASIC-style methodology and using the PlanAhead design and optimization software. Our example includes hierarchical design combined with floorplanning and integrated analysis.

Methodology Used	Timing	Place & Route Runtime	Runtime Reduction
Traditional flattened netlist	-.88 ns negative slack	5.0 hrs.	
With floorplanning and analysis	Meets timing	3.5 hrs.	30%
With incremental ECO capability	Meets timing	40 min.	87%

Table 1 – Methodology results comparison using PlanAhead design and optimization software

In this example, our customer's designers implemented a wireless communications chip with the Virtex™-II 2V6000 platform as their target device. Using a traditional flat methodology, they had 18 timing errors.

By using a block-based, incremental, ECO methodology, their PAR time was 87% faster than their flat methodology. By creating a more efficient floor plan, they were able to meet all of their timing goals. Slice utilization remained a constant 98% throughout.

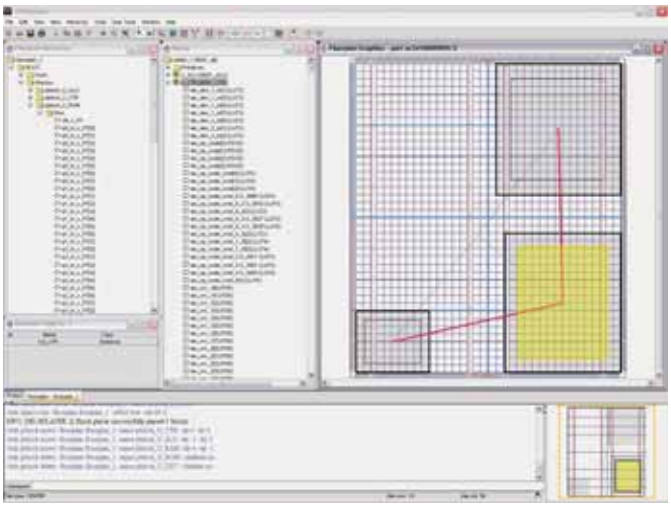


Figure 4 – Place critical timing blocks first.

The first row of Table 1 displays PAR time and design performance using a traditional flattened methodology. The second row demonstrates the benefits of floorplanning. And the third row shows even more benefits by combining floorplanning with an incremental ECO design methodology.

The runtime column lists the total CPU time used by Xilinx PAR. The timing column indicates whether the design was able to meet performance goals.

Tighter Utilization Control

Utilization is an important aspect of FPGA design. In some cases, designers crowd as much logic into a given device as possible to meet production volume requirements.

In other cases, they must leave some space in the FPGA to accommodate bug fixes, naturally occurring design changes, ECOs, or future field upgrades. Leaving room for change is essential for electronic devices with long serviceable life cycles.

Using block-based hierarchical design techniques makes it easier to control utilization. To maximize utilization, you can set the controls at a given utilization level and run a block-level PAR. If it produces successful results, you can increase the utilization setting, which shrinks the block, and then run another block-level PAR.

You can continue this process until the PAR fails, which means you have achieved the maximum possible utilization for the given block. By applying Xilinx area group compression constraints on blocks that are not timing critical, you can also significantly increase utilization.

A wise approach is to leave more space in blocks that have yet to be verified, so you can accommodate anticipated bug fixes.

You might, however, desire a higher utilization setting in blocks that are already field-proven, because it is less likely that extra space will be needed to fix bugs within them.

Teamwork and IP Reuse

Like ASIC designers, you can expedite design time by working as a team and reusing intellectual property (IP) from previous designs. The PlanAhead software provides an ASIC-style hierarchical methodology that facilitates this kind of teamwork.

You can use a block-based approach to divide the work into more manageable blocks and assign responsibilities of designing them to individual team members. You can also reuse blocks (also known as IP cores) from previous designs, or even purchase blocks from a third party to save design and verification time.

When working as such a team, you can begin the physical design process much earlier. You can sequentially implement and assemble the design, beginning with the most critical blocks, as shown in Figure 4. Successively, you can PAR the blocks as designers complete their assignments.

Using ASIC-style methodologies, you can fully characterize design blocks by freezing the placement within them so that power, timing, and other characteristics remain constant. Thus, it's possible to reuse them with consistent results in similar FPGA devices. Because you know that these blocks meet your physical requirements, you can quickly connect them to form larger designs that also meet your requirements.

ASIC designers refer to this type of design as a system-on-chip methodology, and it is particularly valuable if you are an ASIC designer using FPGAs for prototyping. To significantly reduce your overall verification time, use the PlanAhead analysis and floorplanning software to lock placement within pre-characterized IP blocks and connect them together. Because you already know these blocks meet timing and other design requirements, you can create a working ASIC prototype much more quickly.

Conclusion

FPGAs have become so complex that you may be encountering ASIC-size design bottlenecks. You can overcome these problems by adopting proven ASIC-style design methodologies. Hierarchical design enables you to make fast incremental changes, and early analysis allows you to fix problems prior to place-and-route.

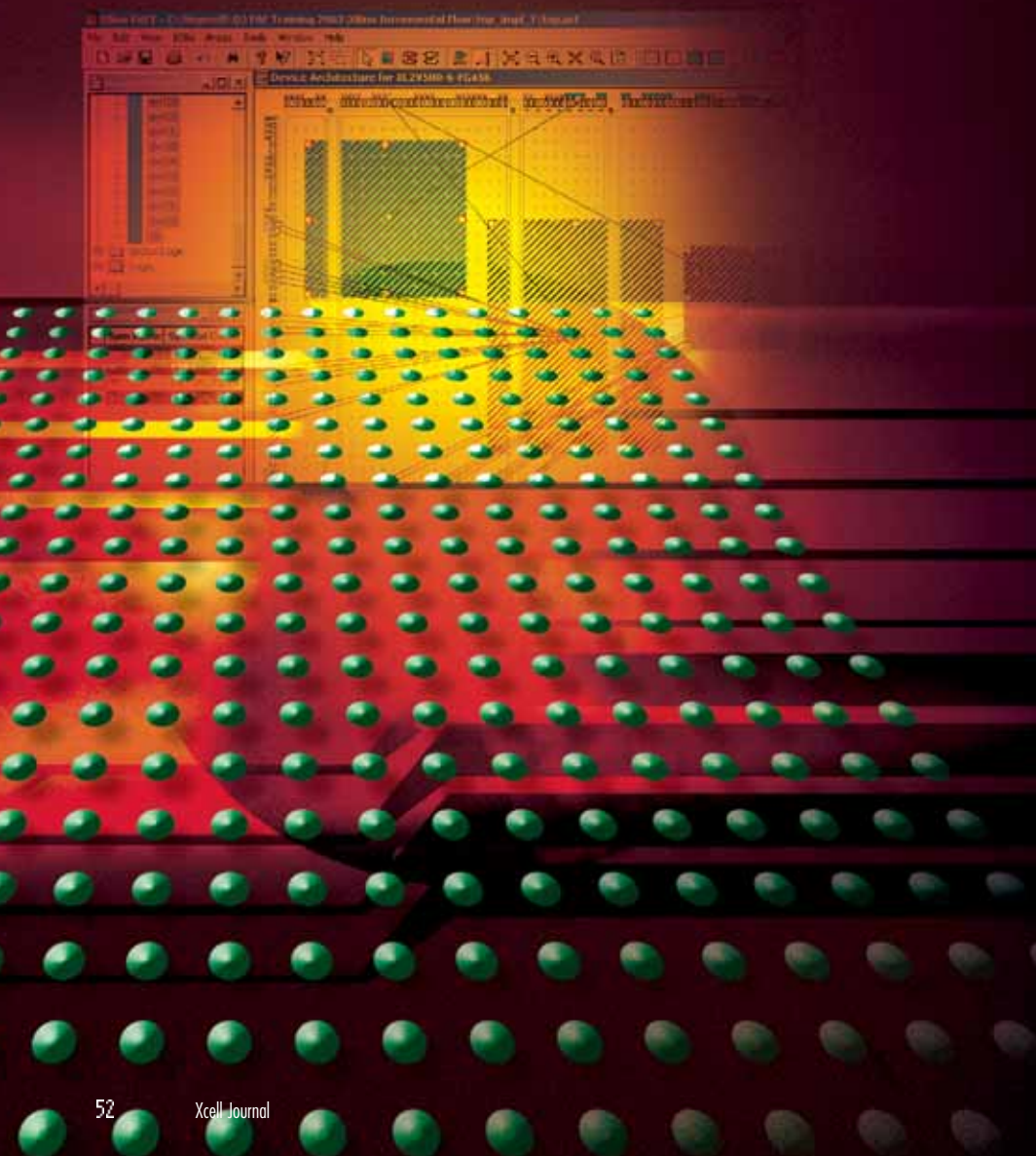
You can closely control utilization, packing designs ever tighter, or leave extra room for bug fixes, ECOs, or planned upgrades for products that have a long field or shelf life.

Finally, you can work as a team and shorten design and verification time by reusing blocks of pre-characterized IP.

To learn more about the PlanAhead hierarchical design and physical optimization software, visit the Hier Design website at *www.hierdesign.com.* Σ

Preserve Timing Gains in Incremental Design

Incremental synthesis tools from Mentor Graphics combined with Xilinx incremental place and route software can reduce your design iteration time.



by Mike Fingeroff
Technical Marketing Engineer, Synthesis Group
Mentor Graphics Corp.
michael_fingeroff@mentor.com

Many months' worth of painstaking synthesis and place and route (PAR) iterations can be lost instantly if you need to make a functional change to your high-performance FPGA design. When ambitious performance goals come into play, achieving timing closure may involve a series of sophisticated synthesis, floorplanning, and PAR steps. Run times of design iterations may take an entire day, so complex designs requiring multiple iterations could add weeks or even months to a design schedule.

You can protect hard-fought timing gains from the disruptive effects of functional iterations through the use of a block-based incremental design methodology. Precision™ RTL Synthesis from Mentor Graphics, used in conjunction with Xilinx ISE software, enables you to perform design iterations at the block level, thus preserving the performance of unaffected blocks. In this article, I will cover the technical details and tradeoffs of using a block-based incremental design methodology.

Incremental Synthesis

Incremental synthesis gives you the ability to make an RTL code change or constraint modification that, when re-synthesized, affects only an isolated portion of your entire design hierarchy. This allows you to reuse the unaffected portion of the technology-mapped synthesis netlist, reducing synthesis runtimes as well as preserving timing on the unchanged design hierarchy.

Design Methodology

Good RTL design practices can significantly improve your design's performance.

- **Functional partitioning of logic –** Partition your design into major hierarchical blocks based on functionality. Although your design blocks can contain sub-hierarchies, the major design partitioning should be done at the top level of the design (Figure 1). Top-level partitioning allows incremental synthesis results to leverage Xilinx incremental PAR techniques. Partitioning your design into groups of functionally related logic also enables the synthesis engine to more efficiently optimize critical portions of your design.
- **Isolation of critical paths –** Isolate critical timing paths into a single block of your design hierarchy. Incremental synthesis preserves timing on unchanged design blocks by disabling hierarchical boundary optimization on blocks that are incrementally re-synthesized. Critical timing paths routed though re-synthesized blocks may lose optimal timing.
- **Registered inputs or outputs on functional blocks –** Register all inputs or outputs of your major hierarchical blocks. This will minimize the effects of disabling hierarchical boundary optimization by preventing long combinatorial logic paths entering or leaving a design block.
- **IOB and clock logic at the top level –** Place all of your I/O logic, buffers, tristates, and clock logic (DCMs, global buffers, and such) at the top level.

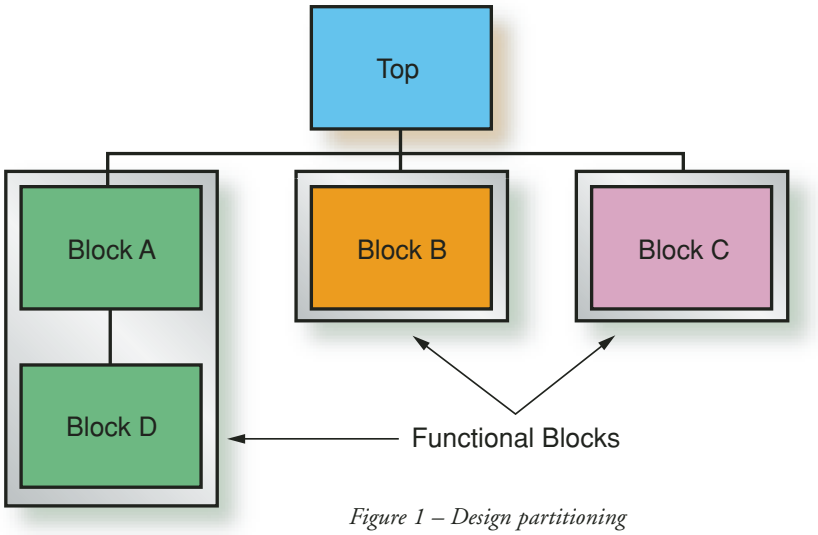


Figure 1 – Design partitioning

Instantiation of I/O buffers in a lower-level design block is acceptable using the Xilinx incremental PAR flow.

Bottom-Up Synthesis Flow

Precision RTL Synthesis supports incremental synthesis using a bottom-up design methodology. After properly partitioning your design, you should create a separate Precision RTL Synthesis project for each major hierarchical block, as well as for the top-level design.

The bottom-up incremental flow can be divided into three steps:

1. **Independent synthesis of major hierarchical blocks –** Synthesize each major hierarchical block as an independent Precision RTL Synthesis project, with I/O insertion disabled. (I/O insertion will be performed later during the top-level assembly.) Precision RTL Synthesis produces both an XDB and EDIF netlist, either of which is valid when assembling the top-level design.
2. **Top-level assembly –** Synthesizing the top-level design stitches together all XDB or EDIF netlists from lower-level design blocks, performs I/O insertion, and does a complete timing characterization of the entire design. The top-level assembly project should consist of a single VHDL, Verilog™, or EDIF file that you can use to instantiate all of the lower-level design

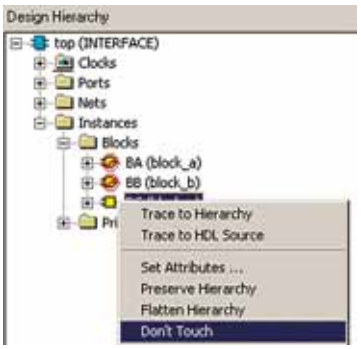


Figure 2 – Applying DONT_TOUCH attributes using the Precision GUI

blocks. You must place DONT_TOUCH attributes on all of the lower-level design blocks before running synthesis to prevent any further optimization. The DONT_TOUCH attributes can be applied either directly in the Precision RTL Synthesis Design Hierarchy Browser (Figure 2) or as an HDL or TCL script attribute.

3. **Making an incremental design change –** Once you have completed and fully synthesized your design, you may wish to make HDL code or constraint modifications to fix bugs or improve performance. In a standard top-down synthesis methodology, this would require you to re-synthesize the entire design. Not only is this time-consuming, but you may introduce timing problems into unmodified portions of your design. However,

using the Precision RTL Synthesis bottom-up methodology for incremental synthesis allows you to isolate the effects of an HDL code change to specific design blocks. Should an HDL code change occur, you are only required to re-synthesize the project that references the modified HDL file. You must then, of course, run a final top-level assembly to generate a new netlist.

Incremental Place and Route

Incremental synthesis helps you to reduce synthesis runtimes and preserve unmodified portions of the technology-mapped netlist. However, you must still depend on the back-end PAR tool to meet or preserve timing goals. You can use the Xilinx ISE incremental PAR flow in conjunction with the Precision RTL Synthesis incremental flow to preserve timing gains by only re-placing those design blocks that have been modified during your incremental synthesis run.

Floorplanning the Design

One of the most critical steps in the Xilinx ISE incremental PAR flow is floorplanning your design. You must perform floorplanning to minimize the interconnect delay between your design blocks. You may also need floorplanning to overcome the performance limitations associated with disabling hierarchical boundary optimization during incremental synthesis.

Floorplanning consists of:

- **Separate area groups for each functional block –** You must create a Xilinx area group for each major design block. Area groups are created and placed using the Xilinx PACE tool, which takes as its input a UCF (produced by Precision RTL Synthesis), and an NGD file produced during the translate stage (Figure 3). Area groups must not overlap, because that might adversely affect performance. Assigning non-overlapping area groups to major hierarchical blocks allows the ISE program to completely re-place and re-route a design block inside an area group when a change is made in the netlist. If possible, you should make the area groups slightly larger than needed. This will allow successful re-placement if your design block utilization increases because of a modification. In addition to assigning area groups, you must also lock down all I/Os.

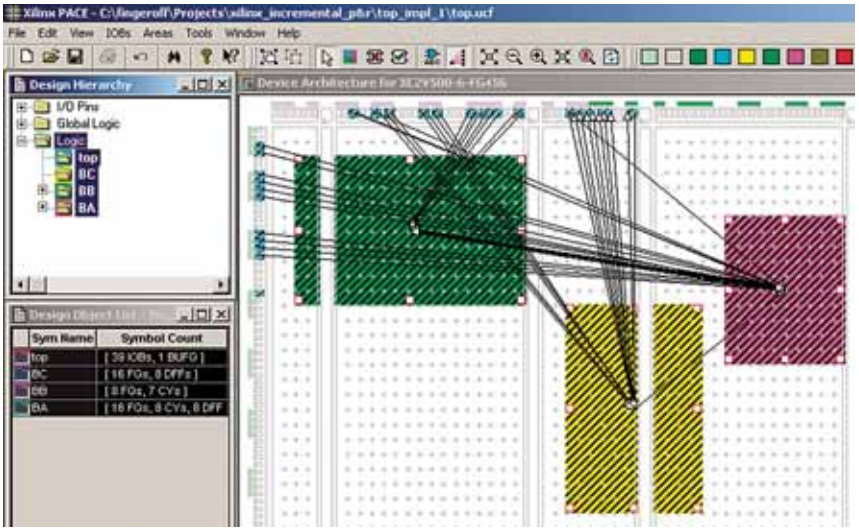


Figure 3 – Floorplanning using Xilinx PACE tool

- **Co-location of related blocks and I/Os –** Not only is it essential to partition your design logically to optimize inter-block timing paths, but it is equally important to arrange the area groups associated with each logical partition to minimize intra-block timing paths. Area groups, as well as I/Os, that communicate with one another should be placed as close as possible to each other.

Running Place and Route

- **Performing initial PAR –** Once you have floorplanned your design by assigning area groups and locking down the I/Os, you must run through a first pass of technology mapping and PAR to generate an initial set of guide files (*.ngm and *.ncd). You can then use the guide files in your next PAR iteration to re-place and re-route as much of your design as possible.
- **Performing guided technology mapping and PAR –** You can enable incremental PAR by setting the guide mode to "incremental" and specifying the guide files produced by the Xilinx technology mapper and PAR from your previous PAR iteration. The guide mode can be specified either in the ISE GUI or at the command line using the -gm switch. The Xilinx incremental guide mode directs the technology mapper and PAR to re-place and re-route all area groups and associated logic that haven't changed, thus preserving timing gains from previous design iterations.

Conclusion

Small design changes can have a devastating effect on your overall system performance, sometimes undoing months' worth of development effort. To avoid these pitfalls, you can use Mentor Graphics Precision RTL Synthesis and Xilinx ISE PAR in tandem to implement an efficient and effective incremental design methodology that preserves timing gains and dramatically reduces your design iteration time.

To learn more about Mentor Graphics Precision RTL Synthesis, visit *www.mentor.com/precision/*.

For a complete description of incremental place and route, see Application Note XAPP418, "Xilinx 5.1i Incremental Design Flow" (search *www.xilinx.com*).

Designing Next-Generation Digital Consumer Electronics Devices

A new handbook guides you through the digital technology maze.

by Xcell Staff

The consumer electronics market is flooded with new products. In fact, the sheer number of consumer devices has introduced entirely new business models for the production and distribution of goods. These new devices have also necessitated major changes in the way we access information and interact with each other.

The digitization of technology (both products and media) has led to leaps in product development. It has enabled easier exchange of media, cheaper and more reliable products, and convenient services.

Digitization has not only improved existing products – it has also created entire new classes of products. For example, improvements in devices such as DVRs (digital video recorders) have revolutionized the way we watch TV. Digital modems bring faster Internet access to the home. MP3 players have completely changed portable digital music. Digital cameras enable instant access to photographs that you can e-mail seconds after taking them, and eBooks allow consumers to download entire books and magazines to eBook tablets. Devices such as wireless phones, personal digital assistants (PDAs), and other pocket-sized communicators are on the brink of a new era in which mobile computing and telephony converge.

This has been made possible through the availability of cheaper and more powerful semiconductor components. With advancements in semiconductor technology, components such as microprocessors, memories, and logic devices are cheaper, faster, and more powerful with every process migration.

Similar advances in programmable logic devices have enabled manufacturers of new consumer electronics devices to use FPGAs and CPLDs to add continuous flexibility and maintain market leadership.

The *Digital Consumer Technology Handbook – A Comprehensive Guide to Devices, Standards, Future Directions, and Programmable Logic Solutions* is a survey of state-of-the-art digital consumer technology and related applications. It provides a broad synthesis of information so that you can make informed decisions about the future of digital appliances.

Organization and Topics

Amit Dhir, author of the *Digital Consumer Technology Handbook*, says that "with so many exciting new digital devices available, it was difficult to narrow the selections to the most important cutting-edge technologies.

"In this book I have included more than 20 emerging product categories that typify current digital electronic trends. The chapters offer exclusive insight into those digital consumer products that have enormous potential to enhance our life and work styles. Each chapter follows a logical progression, from a general overview of a device to its market dynamics to the core technologies that make up the product."

Who Should Read This Book

The Digital Consumer Technology Handbook is meant for anyone seeking a broad-based familiarity with the issues of digital consumer devices. It assumes you have some knowledge of systems, general networking, and Internet concepts.

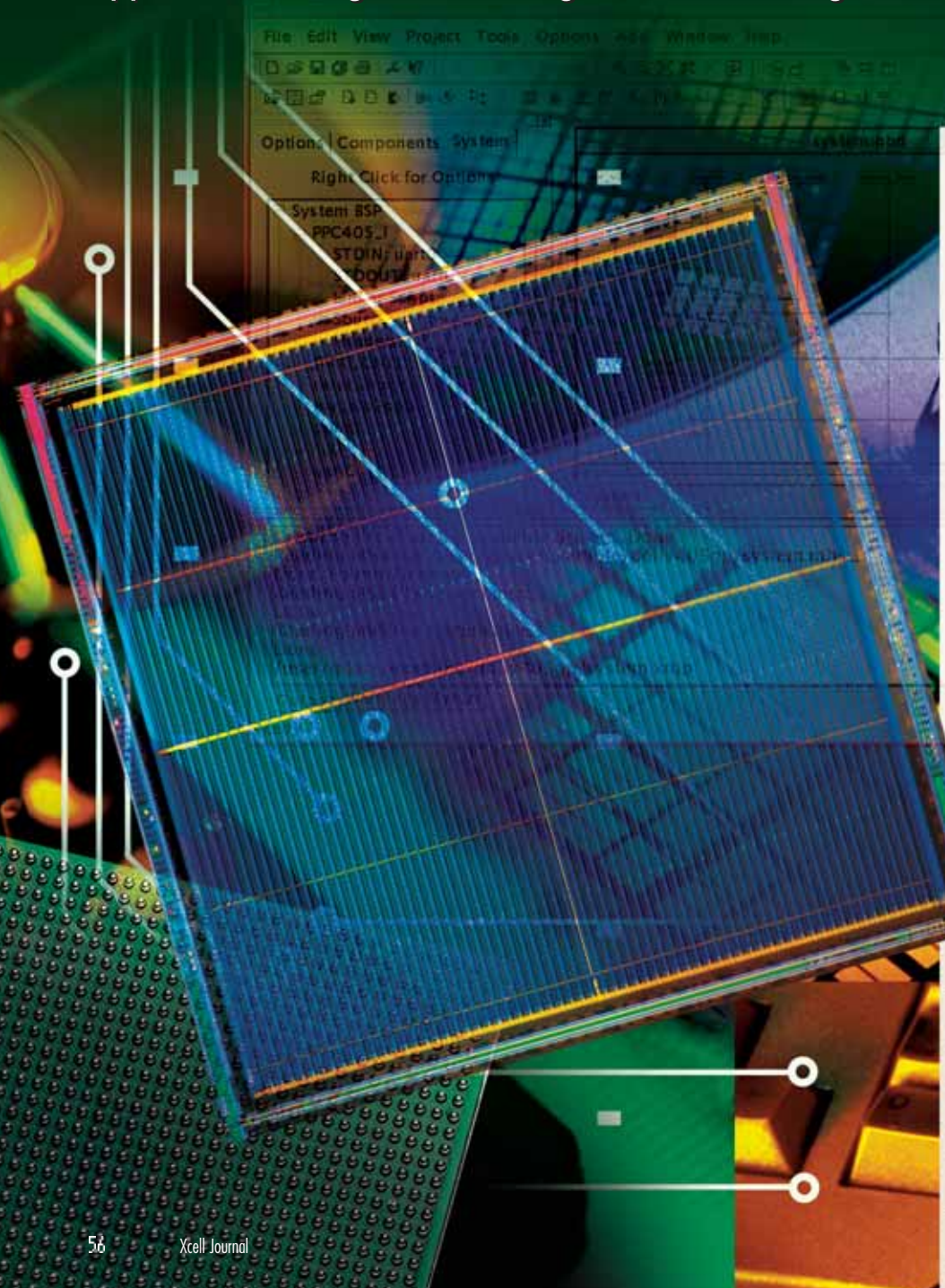
The Digital Consumer Technology Handbook – A Comprehensive Guide to Devices, Standards, Future Directions, and Programmable Logic Solutions is available at all consumer bookstores after March 1, 2004, or online through Elsevier Publishing at *http://books.elsevier.com/computereng/*.

The price of the book is $49.95, but you can receive a special 15% discount off the list price on orders placed before March 1, 2004. Simply enter the offer code 76899 when ordering from the Elsevier website.

Amit Dhir, author of The Digital Consumer Technology Handbook – A Comprehensive Guide to Devices, Standards, Future Directions, and Programmable Logic Solutions

Seamless HW/SW Co-Verification for Xilinx Virtex-II Pro FPGAs

The Mentor Graphics Seamless co-verification tool is as applicable to large FPGA designs as it is to large ASICs.



by Ross Nelson
Technical Marketing Engineer
Mentor Graphics Corporation
ross_nelson@mentor.com

Gary Dare
Technical Marketing Engineer
Mentor Graphics Corporation
gary_dare@mentor.com

As gate counts increase with each new generation of FPGAs, it becomes easier to implement large-scale systems using platform FPGAs rather than ASICs. FPGA design starts have grown along with the technology's capabilities, exceeding those for ASICs by as much as 10 to 1 in 2002.

To address the challenges these large FPGA-centric systems present, we must re-examine system-on-chip (SoC) methodologies, such as hardware/software co-verification, initially developed for ASIC-scale designs.

In this article, we'll demonstrate the Mentor Graphics® Seamless® hardware/software co-verification technology on a design targeting the Xilinx Virtex-II Pro™ FPGA, with its embedded IBM™ PowerPC™ 405 CPU. This design boots up the Nucleus® Plus real-time operating system (RTOS), also from Mentor Graphics. As we'll see, significant performance improvements are possible over conventional HDL simulation when using the Seamless tool.

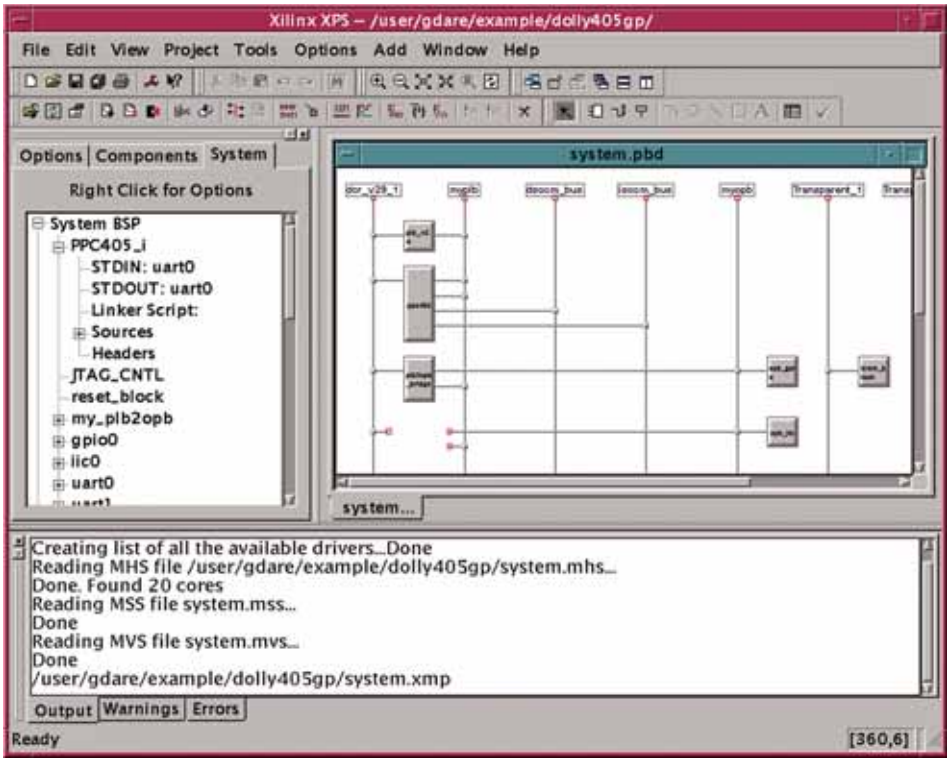


Figure 1 – A schematic of the reference design created with Xilinx EDK tools.

Co-Verification Offers Speed, High Performance

Co-verification is proven in the development of embedded system ASICs. Hardware/software co-verification tools allow you to run software against a hardware design to verify hardware/software interfaces before building a physical prototype. This gives you concurrent access to your hardware design while it is in development and reduces overall project cycle time.

Increased visibility into your hardware design, especially when encountering problems, enables you to debug designs while they're exercised by actual verification software running on the CPU.

From a performance perspective, the behavioral model of the co-verification processor speeds up simulation execution versus a register transfer level (RTL) model of the CPU.

Massive speed gains by as much as 1,000 times are possible by applying Seamless optimizations that remove overhead, such as explicit instruction and data fetch cycles in hardware simulation. This performance improvement enables you to verify large FPGA-based systems with as many as 2 million gates – gates that could not otherwise be verified within a practical time frame using conventional HDL simulation.

With hardware/software co-verification, you can also spot bugs that would otherwise go unnoticed, simply because it would be impractical to test for them. This becomes more important as systems become larger, and the portion of nodes that can be physically probed from the perimeter of the FPGA becomes smaller.

The majority of nodes are internal to the design and observable only through simulation. Even those that can be probed are dependent on cycle time for the design and production of test boards.

Seamless Co-Verification of an FPGA Design

In the following example, you can observe software – whether high-level C or low-level assembler – as well as hardware using the Seamless co-verification tool.

System Specifications

Our example system is built to the scale of typical embedded microcontrollers, such as the IBM PowerPC 405GP, a.k.a. Galaxy. In a Virtex-II Pro design for the Galaxy SoC, we integrated standard IP components around the IBM PowerPC 405 CPU, either from the Xilinx Embedded Design Kit (EDK) library or from third-party providers such as the Mentor Graphics Intellectual Property Division.

At this scale of design, we have:

- Three standard IBM CoreConnect buses: processor local bus (PLB), on-chip peripheral bus (OPB), and device control registers (DCR) bus
- Multiple peripherals: Two GPIO, IIC, Ethernet MAC, direct memory access (DMA) controller, PCI bridge, and memory controllers for Flash, SRAM, and SDRAM
- Support functions such as an interrupt controller and on-chip memory using Xilinx block RAM.

As with typical reference boards, such as the Walnut from IBM for the 405GP, a reference board for the Virtex-II Pro FPGA provides the external memory to support the system.

The hardware design was assembled using Xilinx embedded system tools, as shown in Figure 1.

Once the FPGA hardware design is created, the conversion for co-simulation is quite straightforward. The IBM PowerPC 405 block in the hardware design is replaced by the bus interface model (BIM) component of the IBM PowerPC 405 model from the Seamless processor support package (PSP). Software for the Nucleus Plus RTOS is compiled and loaded to run from Flash memory, which is external to the FPGA as part of the test bench. Hardware/software co-verification can now be performed on this FPGA design, just as it is for ASIC designs.

How Seamless Measures Up

Table 1 outlines the relative performance results from executing the boot up of the Nucleus Plus RTOS of our example system. The baseline figure is a measure of Seamless hardware/software co-verification executing without any optimizations.

The hardware design was created in VHDL to utilize the Xilinx EDK behav-

Mode	Elapsed Wall Clock Time
Verilog SDRAM hardware model	35 minutes
SDRAM memory region optimized, Denali models	30 minutes
SDRAM memory region as software memory	30 minutes
SDRAM region as software memory, optimized for time	15 seconds

Table 1 – Preliminary time trial results for Seamless co-verification of a Xilinx Virtex-II Pro reference design.

ioral model libraries for those components; otherwise Verilog® HDL can be used. Other components not available from the Xilinx libraries were found in the Mentor Graphics Intellectual Property Division's Inventra™ catalog.

The Seamless processor model BIM for the IBM PowerPC 405d5 is in VHDL. The bus is also available in Verilog HDL. Because modern HDL simulators such as the Mentor Graphics ModelSim® application can perform mixed-language simulation, that capability extends to Seamless co-verification for hardware simulation.

The Nucleus Plus RTOS is bundled with the Seamless application, as the software runs separately in the embedded system. For simplicity, it is run from Flash memory at a high address. When using Seamless co-verification on a design, you can assign "software memory" to the Flash region. This feature allows you to start software development while the hardware is evolving.

In this case, development began prior to the final setting of the memory controller for the SRAM, ROM, and Flash memory selected for the system. Likewise, for SDRAM at low addresses in the memory map, you can assign software memory to that region prior to the final settings of the SDRAM controller for the chosen devices.

Our simulation time trials are run in four Seamless modes, all using software memory for Flash (instructions and data) and a mix of scenarios for SDRAM (data) as follows:

A) Verilog SDRAM hardware model

B) SDRAM memory region optimized with Denali models

C) SDRAM memory region as software memory

D) SDRAM memory region as software memory, all memory-optimized for time.

Optimization in Seamless transfers certain bus operations from hardware to the Seamless Coherent Memory Server software. The software executes all of the same instructions and data virtually, from a copy of the memory contents. As a result, bus cycle activity is not needed in the hardware simulation.

Optimization can be activated for instruction fetches only; for set instruction regions (as in case B); or in time (where the software execution on the instruction set simulator [ISS] is asynchronous to the hardware simulation). The preliminary performance findings for our Xilinx platform FPGA design, executing a 1.6 µs segment of the RTOS boot, are shown in Table 1. In further studies, we will port more of the RTOS software to this design. These results will be presented in future articles.

For case A, we had to establish the proper settings of the SDRAM controller for a particular memory device, as well as include sufficient time to initialize both. Were it not for a wait loop (to ensure enough time for SDRAM initialization) and a relatively low level of memory access to the Verilog behavioral model, the elapsed time would have been greater.

Before establishing the controller settings and initialization timing, Seamless allows you to start software development by either optimizing the address space of, for example, a Denali model for the hardware design memory, or by setting that region as software memory.

The results of cases B and C are the same because the solution operates the same from either path. If the software is executing in a CPU-bound manner, then time optimization can yield further performance gains through faster code development, as in D. We can decouple the ISS from the hardware simulation clock when no bus activity is scheduled. Faster software execution and debug can be carried out between hardware activities. Optimization can be turned on and off during the co-simulation session, so breakpoints can be set in the software before executing any hardware-dependent code. This allows system debug of simultaneous hardware and software execution, the key benefit for using Seamless in platform FPGA design.

Conclusion

We have demonstrated that hardware/software co-verification, originally developed for ASIC design, also addresses the challenges of today's largest FPGA designs. Seamless co-verification from Mentor Graphics meets these challenges by providing significant performance improvements over conventional HDL simulation. The Seamless solution continues to be enhanced to provide even greater versatility.

In addition to the performance optimization features discussed here, technological enhancements include:

- Platform-based design tools that speed system design and verification
- A C-based design that raises the level of verification abstraction
- An embedded system optimization tool that ensures performance goals and maximizes design efficiency.

These advances are made even more useful to the FPGA engineer by the growing library of Seamless PSPs, developed in cooperation with leading FPGA vendors such as Xilinx.

To learn more about the Seamless co-verification solution, including information on Mentor Graphics' extensive library of PSPs, please visit *www.mentor.com/soc/verification.*

NALLATECH

The High Performance FPGA Systems Company

Nallatech is the world's leading supplier of high performance FPGA systems. A comprehensive product range delivers the highest performance-density, maximum scalability and most complete solutions available on the market today. Nallatech offers a unique high-speed architecture, combined with support for industry standard PCI, cPCI, PC/104 and VME, along with software support for the most popular operating systems and programming languages. Nallatech provides flexible, easy to implement solutions for Data, Communications, Digital Signal, and Image Processing. Customers benefit from lower costs, reduced power consumption and improved performance in end markets including Telecommunications Infrastructure, Aerospace and Defense, Imaging, and Scientific Computing.

www.nallatech.com/corporate/learnmore.asp

Simplify with Synplicity Synthesis Solutions

Conserve FPGA resources in cost-sensitive designs with Synplicity timing-driven synthesis solutions.

by Steven Elzinga
Product Applications Engineer
Xilinx, Inc.
steven.elzinga@xilinx.com

Often, simply being first to market with an excellent product can assure its success. However, effectively harnessing available resources to allow for less expensive parts – and thus a less expensive overall solution – can also give you a significant advantage over a competitor.

The Synplify™ FPGA synthesis solutions and Synplify Pro™ advanced FPGA synthesis solutions from Synplicity™ have many features that enable you to effectively use the resources in FPGAs. For example, you can set constraints for timing-driven synthesis that may allow you to purchase slower speed grade parts without compromising on performance or quality. The Synplify FPGA synthesis solution and Synplify Pro advanced FPGA synthesis solutions offer a perfect fit for the high-volume, low-cost Spartan™ FPGAs, and the advanced Virtex™ series of FPGAs.

Spartan/Virtex Architecture

Spartan-II series FPGAs are based on the Xilinx Virtex architecture. Each FPGA has a number of clock delay locked loops (CLKDLL) for clock management (each CLKDLL driving one of four global clock lines) and block RAM of 4 Kb each.

Furthermore, the basic Virtex architecture can be broken down into "slices," comprising two RAM-based 4-input look-up tables (LUT), two registers, and dedicated "carry logic" used for arithmetic operations (Figure 1). You can configure the LUT as distributed RAM, a shift register LUT (SRL) with a programmable latency of as many as 16 clock cycles, or an LUT for a 4-input Boolean function.

Inside of every input/output block (IOB) are three flip-flops so that the data coming into and leaving the IOB can be registered, thereby increasing the overall maximum frequency.

Spartan-3 series FPGAs are based on the more advanced Xilinx Virtex-II platform FPGA hardware. The difference is that the CLKDLL has added functionality, along with a total of 16 global clock lines (eight global lines in the Spartan-3). The block RAM is larger, with 18 Kb of memory. Platform FPGAs have more registers in the IOBs to accommodate dual data rate (DDR) applications. The Virtex-II architecture also includes embedded multipliers that can be synchronous or asynchronous.

The Synplify and Synplify Pro tools take advantage of the hardware present in the Spartan and Virtex families to ensure optimal area and speed results.

Timing-Driven Synthesis

The Synplify Pro solution is an advanced synthesis tool that performs timing optimizations based on the timing constraints entered. Optimizations are based on – but not limited to – offset and period constraints. Timing constraints are entered in a constraints file (text or GUI entry) or through a tool command language (TCL) script.

Because Synplify Pro software is timing driven, you must provide realistic timing constraints. Avoid over-constraining by asking for more than 10% or 15% of what you really need; in other words, if you need 100 MHz, enter 100 MHz or perhaps 110 MHz. The timing-driven aspect of the Synplify Pro software first meets your timing constraints and once met, minimizes area. This allows you to achieve your performance goal while using the smallest, lowest cost device.

Synplicity's SCOPE® (synthesis constraints optimization environment) GUI allows you to easily manipulate constraints from the register transfer level (RTL) viewer into the spreadsheet. Just click and drag any object into the spreadsheet and apply the appropriate constraint, or select the objects from pull-down menus in the spreadsheet's individual cells (Figure 2).

The SCOPE interface provides a single location for both synthesis and place-and-route (PAR) constraints, which are passed along to the Xilinx PAR tool. Clocks, inputs, and outputs are all automatically loaded into the SCOPE utility for fast and easy constraint entry.

The Synplify Pro synthesis tool has options that allow you to manipulate regis-

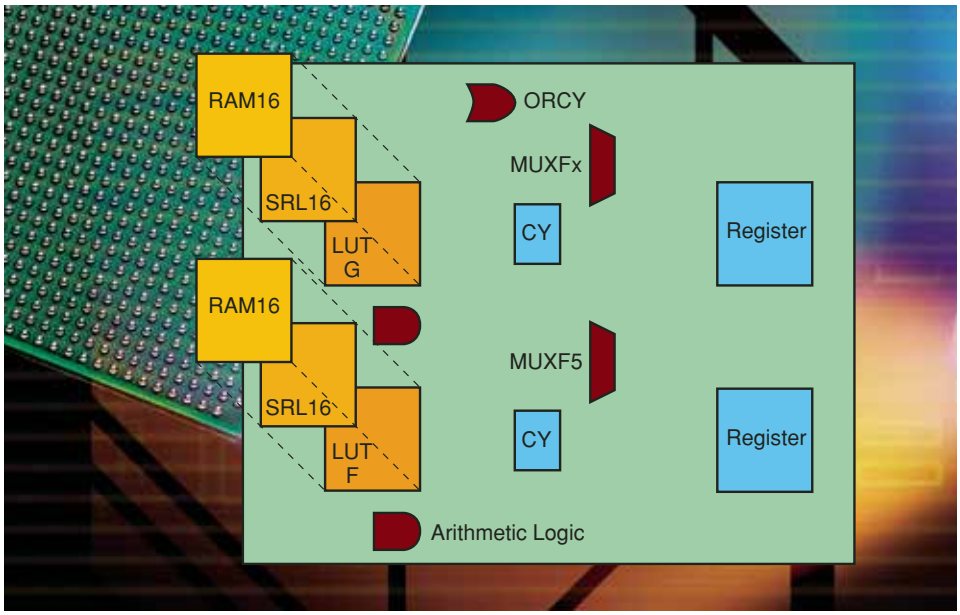


Figure 1 – Virtex slice

ters and increase the maximum frequency in your design. One such option is retiming. With the retiming switch on, the Synplify Pro program moves existing registers through combinatorial logic to balance timing delay between registers. This option is very helpful when your design is pipelined.

Other features include the inference of SRLs (which saves resources) and IOB flip-flops: You can control both through the use of the synthesis directives syn_srlstyle and syn_useioff.

Synplify Pro software also includes FSM Explorer and FSM Compiler. With FSM Explorer, the Synplify Pro synthesis tool locates the design's finite state machines (FSM) and determines the optimal encoding style for the FSM to meet timing constraints.

To explore different state machine encoding styles, just turn off FSM Explorer and select the encoding style by using the syn_encoding directive, along with the choices default, one-hot, gray, sequential, and safe. When selecting default, a predetermined encoding style will be used based on the number of states in the FSM.

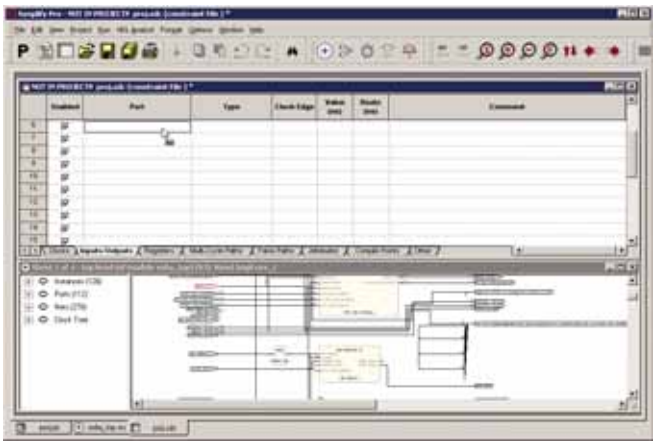


Figure 2 – Drag-and-drop constraint entry in Synplify Pro SCOPE?

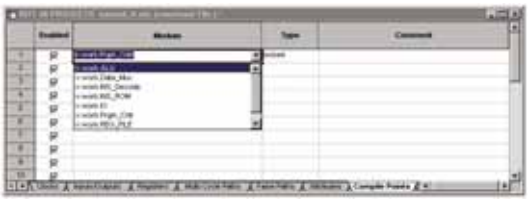


Figure 3 – Synplicity MultiPoint setup

Incremental Design

Synplicity's MultiPoint™ technology allows you to work incrementally on individual modules in your design – without having to re-synthesize the entire design each time you make a change. The MultiPoint flow provides a superior approach to incremental design that provides stability during design changes without compromising the quality of results. You can set the compile points in the constraints file by manual entry or through the SCOPE interface (Figure 3).

The MultiPoint technology is intelligent, and knows when logic changes require recompilation. For example, an HDL syntax change or an added comment will not trigger recompilation, but a change that alters the logic will.

You can use MultiPoint synthesis in conjunction with the incremental design flow process in the Xilinx implementation tool, along with "cross probing" to better achieve timing closing.

Entering realistic timing constraints is important, because once timing constraints are met, the Synplify Pro tool works on area optimizations. If your critical path does not go through your state machine, it's easy to try a different encoding style that may not be as efficient in timing but is more efficient in area, freeing up resources in the critical path.

Synplify Pro software further optimizes your HDL code with its numerous inference templates. With inference templates, the Synplify Pro program takes advantage of the resources available in FPGAs. Depending on how you set certain synthesis directives, the Synplify software infers SRLs, block multipliers, block RAM, and ROM in block RAM, saving LUTs for the combinatorial logic.

Conclusion

In today's economic climate, cost savings are paramount. Together, Synplicity and Xilinx offer a combination of synthesis software optimized for timing, area, and cost savings on the best price-for-performance FPGAs available. To see what the Synplify and Synplify Pro synthesis tools can do for your Xilinx FPGA design, download free, fully functional evaluation copies of the Synplify and Synplify Pro solutions from *www.synplicity.com/downloads/*. For additional information on the Spartan-3 family of FPGAs, visit *www.xilinx.com/Spartan3/*. To learn more about Virtex FPGAs, go to *www.xilinx.com/virtex2/*.

Put the Right Bus in Your Car

The amazing array of features available in today's cars has spawned new in-vehicle bus standards.

by Karen Parnell
Automotive Product Manager
Xilinx, Inc.
karen.parnell@xilinx.com

The next few years will be a rocky road for automobile electronics designers. No single in-car data bus can adequately handle the entertainment, safety, and intelligent-control requirements of the cars that will roll off assembly lines in North America, Europe, and Asia.

Choosing the right data bus can lead to a competitive advantage, but the selection is increasingly difficult, as carmakers around the globe adopt different solutions.

Electronics Drives Innovation

In-car electronics have grown tremendously in recent years. Traditional body-control and engine-management functions, plus new driver-assistance and telematics systems, have spurred annual growth rates as high as 16%, according to the Institute of Electrical and Electronics Engineers (IEEE). The IEEE forecasts that electronics will account for 25% of the cost of a mid-size car by 2005.

One high-growth area is telematics systems – the convergence of mobile telecommunications and information processing in cars. Significantly, telematics applications exhibit market characteristics similar to those of consumer products: short time to market, short time in market, and changing standards and protocols. These market characteristics are just the opposite of the relatively long design cycles of traditional in-car electronics, which are often dictated by safety and tooling-cost considerations.

Traditional systems such as CAN (controller area network) and J1850 have been used in body control for many years. But bandwidth and speed restrictions make it difficult for these serial, event-driven buses to handle newer real-time applications.

A number of new bus standards have emerged featuring time-triggered protocols and optical data buses. These in-car bus networks can be divided into four categories:

- Body control – dashboard/instrument panel clusters, mirrors, seat belts, door locks, and passive airbags
- Entertainment and driver-information systems – radios, Web browsers, CD/DVD players, telematics, and infotainment systems
- Under the hood – antilock brakes, emission control, power train, and transmission systems
- Advanced safety systems – brake-by-wire, steer-by-wire, and driver assistance systems (active safety).

Figure 1 shows the relative speeds of the various bus systems, which range from kilobits per second to gigabits per second.

"Under-the-Hood" Buses

Two networks found under the hood serve functions ranging from seat adjustment to antilock brakes.

Controller Area Network

One of the first and most enduring control networks, the CAN bus, is the most widely used, with more than 100 million nodes installed worldwide.

A typical vehicle integrates two or three CAN buses operating at different speeds. A low-speed CAN bus runs at less than 125 Kbps and manages body-control electronics, such as seat and window movement controls and other simple user interfaces. A high-speed (up to 1 Mbps) CAN bus runs real-time critical functions such as engine management, antilock brakes, and cruise control.

CAN protocols are becoming standard for under-the-hood connectivity in cars, trucks, and off-road vehicles. One outstanding feature of the CAN protocol is its high transmission reliability.

Local Interconnect Network

The local interconnect network (LIN) was developed to supplement the CAN bus in applications where cost is critical and data transfer rates are low. The LIN bus is an inexpensive serial bus used for distributed body control electronic systems. It enables effective communication for smart sensors and actuators where the bandwidth and versatility of the CAN bus are not required. Typical applications are door control (windows, door locks, and mirrors), seats, climate regulation, lighting, and rain sensors.

The LIN bus is a UART-based, single-master, multiple-slave networking architecture originally developed for automotive sensor and actuator networking applications. The LIN master node connects the LIN network with higher-level networks such as a CAN bus, extending the benefits of networking all the way to the individual sensors and actuators.

Entertainment and Driver Information Systems

Car infotainment and telematics devices, especially car navigation systems, require highly functional operating systems and connectivity. Until now, both open-standard and proprietary standalone buses have coexisted independently and peacefully. But because of the pressures of convergence, future systems will require integrated electronic subsystems.

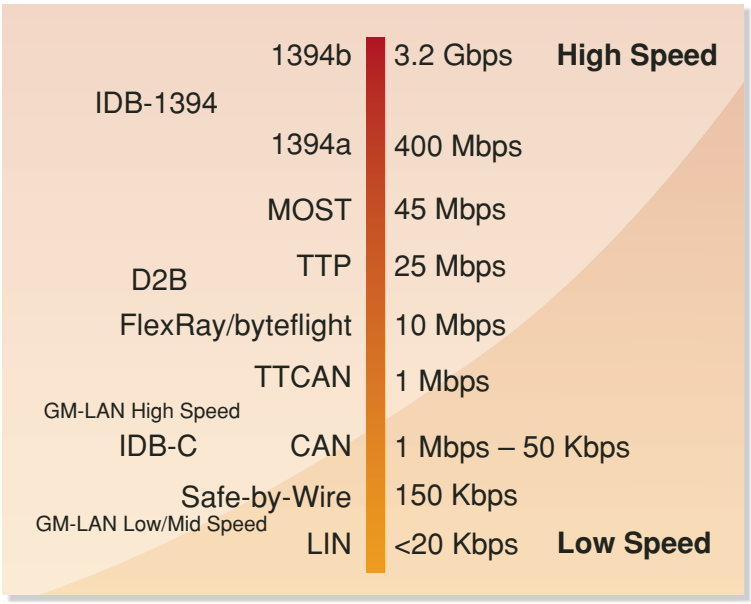


Figure 1 – In-car network data-transfer speeds

By relying on open industry standards, all key players – from manufacturers to service centers to retailers – can focus on delivering core expertise to the customer. Open standards will save the duplication of time and effort it would take to develop separate, incompatible designs for specific vehicles or proprietary computing platforms.

Several organizations and consortia are leading standardization efforts, including the MOST (Media Oriented System Transport) Cooperation, the IDB (Intelligent Transport System Data Bus) Forum, and the Bluetooth™ Special Interest Group (SIG).

Media Oriented System Transport

MOST networks connect multiple devices, including car navigation, digital radios, displays, cellular phones, and CD/DVDs. MOST technology is optimized for use with plastic optical fiber. It supports data rates as high as 24.8 Mbps and is highly reliable and scalable at the device level.

MOST offers full support for real-time audio and compressed video. It is vigorously supported by German automakers and suppliers. The MOST bus is endorsed by BMW, DaimlerChrysler, Harman/Becker, and OASIS Silicon Systems. A recent notable example of a MOST implementation is its use by Harman/Becker in the latest BMW 7 series.

Intelligent Transport System Data Bus

The IDB Forum manages the IDB-C and IDB-1394 buses and standard interfaces for OEMs that develop aftermarket and portable devices. Based on the CAN bus, IDB-C is geared toward devices with data rates of 250 Kbps.

IDB-1394 (based on the IEEE-1394 FireWire™ standard) is designed for high-speed multimedia applications. IDB-1394 is a 400 Mbps network using fiber-optic technology. Applications include DVD and CD changers, displays, and audio/video systems.

IDB-1394 also allows 1394-portable consumer electronic devices to connect and interoperate with an in-vehicle network. Zayante Inc., for example, supplies 1394 physical layer devices for the consumer market. A recent joint demonstration with the Ford Motor Company included plug-and-play connections of a digital video camera and a Sony PlayStation™ 2 game console, as well as two video displays and a DVD player.

Digital Data Bus

The Digital Data Bus (D2B) is a networking protocol for multimedia data commu-

nication that integrates digital audio, video, and other high-data-rate synchronous or asynchronous signals. It can run as fast as 11.2 Mbps and be built around either SmartWire™ unshielded twisted pair cable or a single optical fiber.

This communication network is being driven by C&C Electronics in the UK and has industry acceptance from Jaguar and Mercedes-Benz. The integrated multimedia communication systems deployed in the Jaguar X-Type, S-Type, and new XJ Saloon, for example, use D2B.

The D2B optical multimedia system is designed to evolve in line with new technologies while remaining backwards compatible. D2B optical is based on an open architecture that simplifies expansion, because changes to the cable harness are not required when adding a new device or function to the optical ring. The bus uses just one polymer optical fiber to handle the in-car multimedia data and control information. This gives better reliability, fewer external components and connectors, and a significant reduction in overall system weight.

Bluetooth and ZigBee

Bluetooth wireless technology is a low-cost, low-power, short-range radio protocol for mobile devices and WAN/LAN access points. Its specification describes how mobile phones, computers, and PDAs can easily interconnect with each other, with home and business phones, and with computers.

The Bluetooth SIG includes such members as AMIC, BMW, DaimlerChrysler, Ford, General Motors, Toyota, and Volkswagen. An example of Bluetooth deployment in cars is Johnson Controls' BlueConnect™ technology, a hands-free system that allows drivers to keep their hands on the wheel while staying connected through a Bluetooth-enabled cellular phone.

There is, however, some concern about long-term support of Bluetooth devices. The problem centers on how the electromagnetically noisy in-car environment will affect Bluetooth operation. The lifecycle of cars and other vehicles is much longer than that for consumer products or mobile phones, so silicon manufacturers must address this mismatch between support and service timescales. On the other hand, Chrysler showed Bluetooth connectivity in its vehicles at Convergence 2002.

Some feel Bluetooth technology may be overkill in the car environment, however. So, an emerging standard for low data rate wireless data transfer and control has entered the scene. The ZigBee™ wireless networking solution is a low-data-rate (868 MHz to 2.4 GHz), low-power, low-cost system pioneered by Philips. The ZigBee range is up to 75 meters and is equally at home in industrial control, home automation, consumer, and possibly automotive applications.

Advanced Safety Systems

Safety equipment has evolved from the physical to the electronic domain, starting with advancements in tire and braking technology, through side impact protection and airbags, and on to today's driver-assistance systems.

The latest vehicles are electronics-rich and sensor-based to continuously evaluate the surroundings, display relevant information to the driver, and, in some instances, even take control of the vehicle.

Advanced safety systems include by-wire (for example, drive-by-wire and brake-by-wire), which will replace traditional hydraulic and mechanical linkages with safer, lighter electronic systems.

Other examples of advanced, real-time safety systems include distance control, self-adjusting and sensing airbag systems, radar parking, reversing aids, and back guide monitors (cameras set in the car's bumpers to aid parking).

FlexRay

The FlexRay™ network communication system is aimed at the next generation of by-wire automotive applications. These applications demand high-speed buses that are deterministic, fault-tolerant, and capable of supporting distributed control systems.

BMW, DaimlerChrysler, Philips Semiconductors, Motorola, and the newest member, Bosch, are developing the FlexRay standard for next-generation applications.

The FlexRay system is more than a communications protocol. It also includes a specially designed high-speed transceiver and the definition of hardware and software interfaces between various components of a FlexRay "node." The FlexRay protocol defines the format and function of the communication process within a networked automotive system. It is designed to complement CAN, LIN, and MOST networks.

As a scalable system, FlexRay technology supports both synchronous and asynchronous data transmission. The synchronous data transmission enables time-triggered communication to meet the requirement of dependable systems. FlexRay's synchronous data transmission is deterministic, with guaranteed minimum message latency and message jitter. It supports redundancy and fault-tolerant distributed clock synchronization to keep the schedule of all network nodes within a tight, predefined, precision window.

The asynchronous transmission, based on the fundamentals of the byteflight™ protocol, allows each node to use the full bandwidth for event-driven communications.

Time-Triggered Protocol

Designed for fault-tolerant, real-time distributed systems, the time-triggered protocol (TTP) ensures that there is no single point of failure.

TTP is a mature, low-cost solution that can handle safety-critical applications. Second-generation silicon supporting communication speeds as high as 25 Mbps is available today. The TTA Group, the governing body for TTP, includes Audi, SA, Renault, NEC, TTChip, Delphi, and Visteon among its members.

Time-Triggered CAN

The time-triggered CAN (TTCAN) standard is an extension of the CAN protocol. It adds a session layer on top of the existing data link layer and physical layers to ensure that all transmission deadlines are met, even at peak bus loads. The protocol implements a hybrid, time-triggered, TDMA

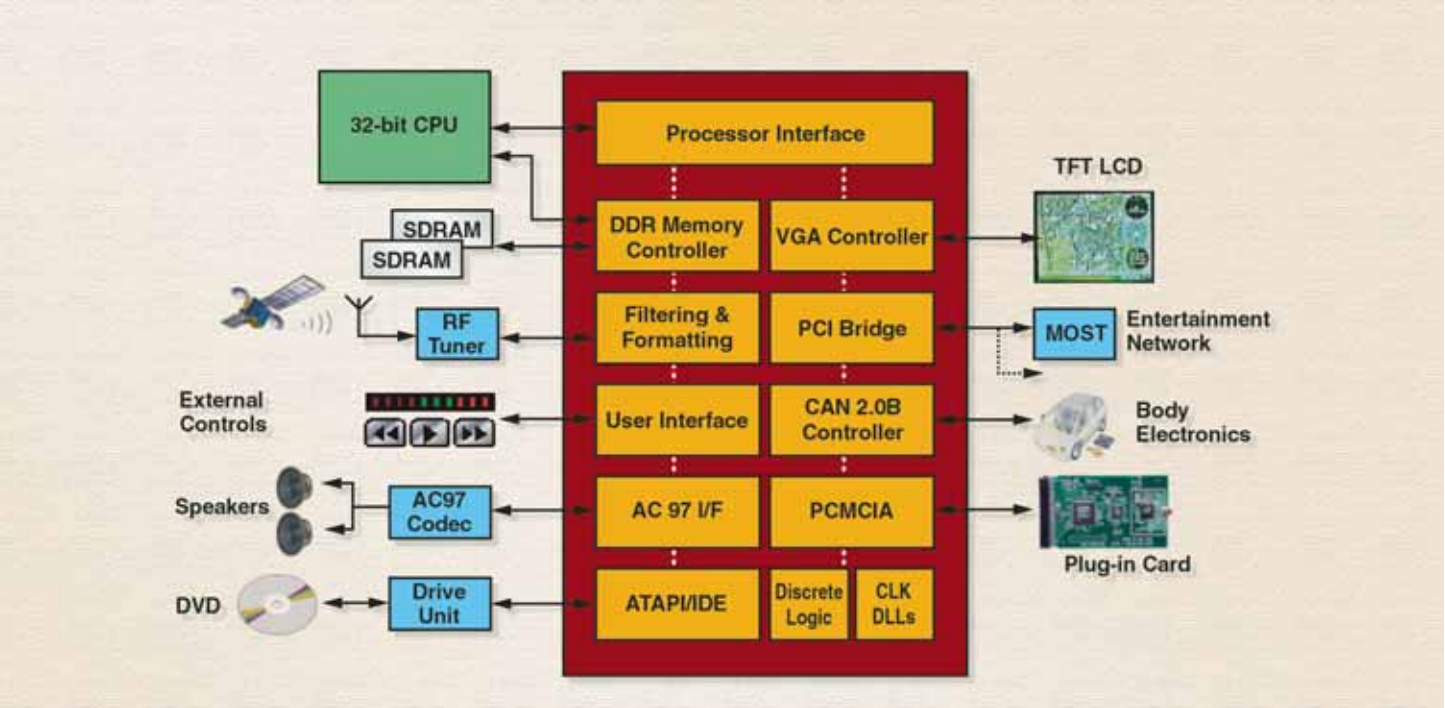


Figure 2 – In-car multimedia functions embedded in an FPGA

(time-division multiplexed access) schedule that also accommodates event-triggered communications. Some of the intended TTCAN uses include engine management systems and transmission and chassis controls, with scope for by-wire applications.

The Programmable Logic Solution

As we have seen from the proliferating number of in-car bus standards, the next few years will become a minefield for automotive electronics designers. Choosing the right data bus will be crucial to success – now measured not simply during integration and testing of units for production, but long after the car has rolled off the assembly line.

The problem is amplified for Tier 1 suppliers and aftermarket companies that supply units to many OEMs, because these customers are likely to opt for different data buses and protocols. The industry has seen a huge shift away from designing a different unit for every OEM – indeed for every car model. Taking its place is a design philosophy that emphasizes reconfigurable platforms.

Design platforms that are cleverly partitioned between software and reprogrammable hardware let manufacturers change system buses and interfaces late in the design process – and even in production. The reconfigurable system concept supports try-outs of different standards and protocols. Programmable logic devices (PLDs) in the form of FPGAs and CPLDs enable modification during all phases of design – from prototype through pre-production and into production.

PLDs can also alleviate over-stocking and inventory issues, because generic FPGAs can be used across many projects and are not application-specific. Once the programmable logic-based unit is on the road, it can even be reconfigured remotely via a wireless communication link to allow for system upgrades or extra functions.

Drop-In IP Cores

The reconfigurable hardware platform can be brought to market quickly by utilizing drop-in intellectual property (IP) core blocks. Memec Design, for example, recently announced the availability of a cost-optimized CAN core interface that includes the complete data link layer, including the framer, transmit-and-receive control, error core design, and flexible interface. Bit rate and sub-bit segments can be configured to meet the timing specification of the connected CAN bus.

The Memec core is designed to provide a bus bit rate of up to 1 Mbps, with a minimum core clock frequency of 8 MHz. It can provide an interface between the message filter, the message priority mechanism, and various system functions such as sensor/activator controls.

Alternatively, the Memec CAN bus can be embedded into a system application interfacing with the microprocessor and various peripheral functions. Another example is Intelliga's iLIN™ core, which is supplied as a LIN bus controller IP core. It uses a synchronous 8-bit general-purpose microcontroller interface with minimal buffering to transport message data. In addition, the reference design includes a single slave message response filter and a software interface that allows the connected microcontroller to perform address filtration.

This emerging LIN body control protocol can be easily tried and tested using the Intelliga iDEV prototyping board, which can demonstrate not only LIN but also CAN and TTCAN buses – all implemented in a Xilinx Spartan™-IIE FPGA.

Figure 2 shows a generic in-car multimedia system design with a CAN core supplying communications for a PCMCIA

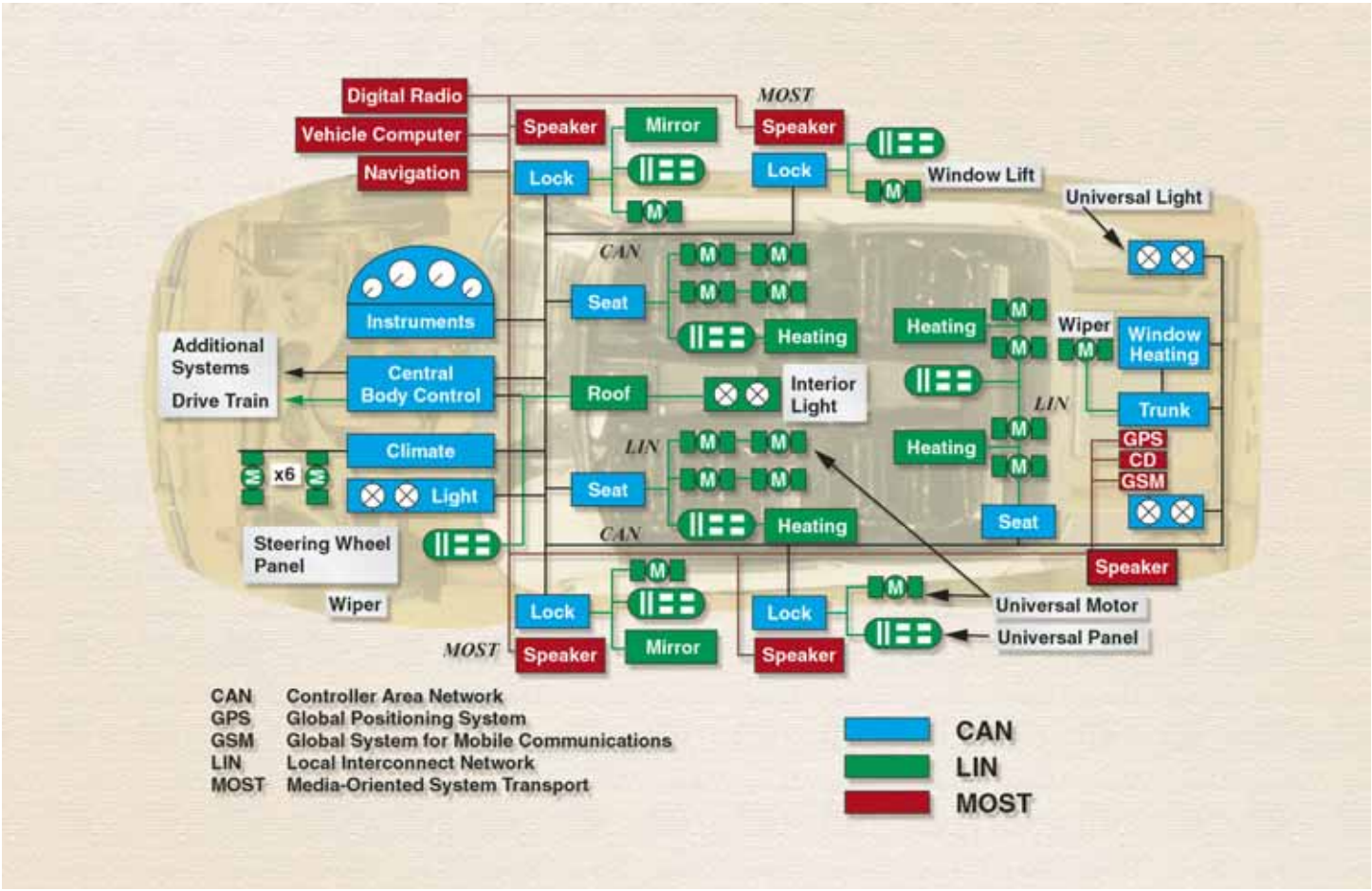


Figure 3 – In-car complementary networks

interface, PCI bridging, an IDE interface, and other functions. One printed circuit board can be used for many customers with customization in the FPGA instead of the board. The model can be extended to include modification or upgrades in the field via wireless connection to reconfigure the FPGA in-system.

Bus Coexistence

Several in-car bus networks can coexist to deliver the right combination of data rates, robustness, and cost.

Figure 3 shows the LIN bus handling low-cost, low-speed connections between the motors for the mirrors, roof, windows, and so forth. A CAN bus handles data communication and control between the instrument cluster, body controllers, door locks, and climate control. Finally, the high-speed MOST optical bus connects the entertainment, navigation, and communication devices.

This model can be extended to include a FlexRay bus (or other real-time safety bus) to handle the high-speed, real-time data required by advanced safety systems.

Conclusion

The delicate engineering balance between cost, reliability, and performance has created a variety of emerging in-car bus systems that will complicate design decisions for years to come. Therefore, automotive OEMs are backing more than one standard due to uncertainties over which one will eventually prevail.

There is an elegant solution to this dilemma. Reconfigurable platforms based on Xilinx programmable logic and IP cores are the best way out of this design predicament. Without sacrificing performance or cost advantages, reconfigurable hardware and software systems from Xilinx allow manufacturers to quickly accommodate changing standards and protocols late in the design process, in production, and even on the road.

Website resources	
CAN	*www.can.bosch.com*
GM LAN	*www.gmtcny.com/lan.htm*
MOST	*www.mostnet.de*
TTTech (TTP)	*www.ttagroup.org, www.ttchip.com*
FlexRay	*www.flexray.com*
D2B	*www.candc.co.uk*
LIN	*www.lin-subbus.org*
Intelliga (LIN)	*www.intelliga.co.uk*
Bluetooth	*www.bluetooth.com*
IDB	*www.idbforum.org*
ZigBee	*www.zigbee.org*
Xilinx Automotive	*www.xilinx.com/automotive/*

A Brand New Creation!

SUNDANCE

Xilinx & FPGA TI DSP

Adding Flexibility to power

SMT365E DSP & FPGA Module
- 64 bits 600MHz TMS320C6416 DSP
- Xilinx Virtex II FPGA XC2V6000
- 256MB of SDRAM @100MHz and 8MB of FLASH
- Six 400Mbytes/s Sundance High-speed Bus
- Six ComPorts for InterDSP communications

SMT374 DSP & FPGA Module
- Two 225MHz TMS320C6713 Floating Point DSP
- Xilinx Virtex II FPGA XC2V2000-4
- 256MB of SDRAM @100MHz and 8MB of FLASH
- Six ComPorts and two 400MB/s SHB ports

SMT398 FPGA Module
- Virtex II range of FPGA devices
- Up to 8 million gates as a re-configurable resource
- 4MB of high speed ZBT SRAM,
- 2MB of high speed QDR-SRAM,
- External Clock & LVDS Interface
- Multiple FPGA/DSP support

SMT406 FPGA PMC Module
- Virtex-II FPGA from V4000 to V8000
- 128MB of 133Mhz SDRAM, accessible from FPGA and PCI bus
- 16MB, 200MHz DDR SRAM or QDR SRAM
- 32MB, 250MHz DDR SDRAM and 16MB of Flash
- 64 bits, 66MHz PCI interface
- 4 programmable clocks generating up to 500MHz

SUNDANCE DIGITAL SIGNAL PROCESSING INC. Tel: (775) 827-3103 USA

SUNDANCE MULTIPROCESSOR TECHNOLOGY LTD. Tel: +44 (0) 1494-793167 UK

SUNDANCE ITALIA S.R.L. Tel: +39 0185 91055 ITALY

Email: sales@sundance.com http://www.sundance.com

Verify Gateway Designs for Time-Triggered Networks

Xilinx programmable logic is ideal for communicating across time-triggered networks.

by Shehryar Shaheen
Research Engineer
PEI Circuits and Systems Research Centre,
University of Limerick
shehryar.shaheen@ul.ie

For future automotive in-vehicle control applications, time-triggered networks will provide the backbone for reliable fault-tolerant communications among control network nodes.

Although the automotive industry is pushing towards an agreement on a single control network standard, don't assume that one single standard will dominate. There will still be a need to communicate across time-triggered networks with different specifications and protocols – a type of gateway.

However, because some of these protocols are still under review, developing and testing such a gateway design with fixed interfaces cannot be future-proof. The gateway core and its associated test bed must be configurable and flexible to accommodate future changes.

This is an ideal situation to exploit the power and flexibility of programmable logic. As researchers at PEI Circuits and Systems Research Centre (CSRC) at the University of Limerick in Ireland, we have developed a Xilinx FPGA-based configurable frame generator/analyzer that can verify a gateway design for time-triggered networks, for real-time applications.

Architecture

The Frame Generator and Frame Analyzer are two distinct designs. Both are synthesized together and placed in a Xilinx Spartan™-II XC2S100 FPGA on an Insight™ Electronics Spartan-II development board.

Frame Generator

The Frame Generator generates frames conforming to a proprietary byte-wide protocol. The generator has the following sub-modules:

- Transmit interfaces
- Frame generation logic
- Frame Generator state machine
- Frame sets.

Figure 1 shows the building blocks of the Frame Generator.

The Frame Generator's four transmit interfaces emulate the function of sending frames to the gateway from a time-triggered communication controller's host controller. Each transmit interface comprises three control signals and an 8-bit port for byte-wide transmission of the frame. The three control signals are:

- Frame Start Out
- Port Ready
- Frame End Out.

Frame Start Out signals the transmission of a new frame.

On each rising edge of Port Ready, the most significant byte (MSB) of the transmitting frame is sent to the 8-bit output port.

On the falling edge of Port Ready, the receiving side reads in the MSB of the frame. The whole frame is broken down into bytes like this and transmitted.

When frame transmission is complete, a Frame End Out pulse is sent to the receiver to indicate an end-of-frame transmission so that the receiver then takes the appropriate action, which is explained later.

The Frame Generator state machine is a 13-state finite state machine that controls the frame generation logic based on its present state.

The frame generation logic block is responsible for generating the right timing and picking up the next frame to be transmitted from the set of frames as defined in the four frame sets.

The frame sets are the offline-defined set of frames to be transmitted by the Frame Generator.

Frame Analyzer

The Frame Analyzer receives frames conforming to a byte-wide proprietary protocol and stores them in message RAMs for analysis.

The Frame Analyzer has the following sub-modules:

- Receive interfaces
- Xilinx block RAMs
- 4 x 1 dual inline package (DIP)-switch controlled demultiplexer (DMUX)
- QuickLogic™ RS 232 core.

Figure 2 illustrates the construction of the Frame Analyzer.

The Frame Analyzer's four receive interfaces emulate the function of receiving frames from the gateway by a time-triggered communication controller's host controller. Each receive interface receives frames according to a proprietary byte-wide protocol. The receive protocol has three control signals and one 8-bit reception port.

The control signals for byte-wide frame reception are:

- Frame Start In
- Ready to Read
- Frame End In.

A low to high transition on Frame Start In signals the arrival of a new frame. The transmitter will make the MSB of the transmitting frame available on the rising edge of its Port Ready.

Port Ready is connected to Ready to Read at the receiver end. On the falling edge of Ready to Read, the byte is read into the Frame Analyzer.

Once all of the bytes making up the frame are received, the Frame Analyzer waits for a Frame End In pulse. A low to high transition on the Frame End In signals the end of a frame reception.

The dual-port block RAMs of the XC2S100 Spartan-II device are used as buffers for storing frames. As the frames are received on each of the four receive interfaces, they are written into the four block RAMs associated with each interface. This is the first phase.

Once the frame reception is complete, the mode of the Frame Analyzer is changed from "write" to "read."

In the second phase, the frames are read out of the FPGA though an RS-232 connection to a PC. The block RAMs for the Frame

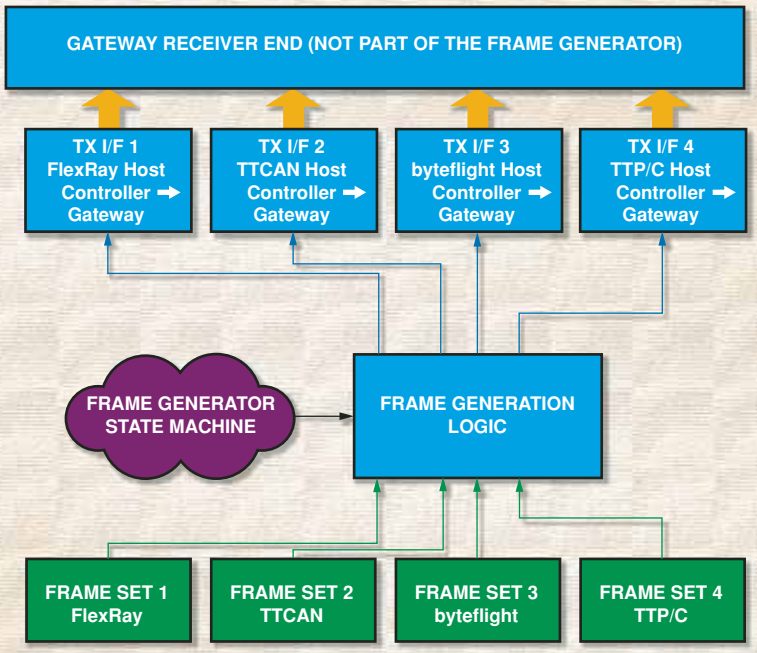


Figure 1 – Frame Generator building blocks

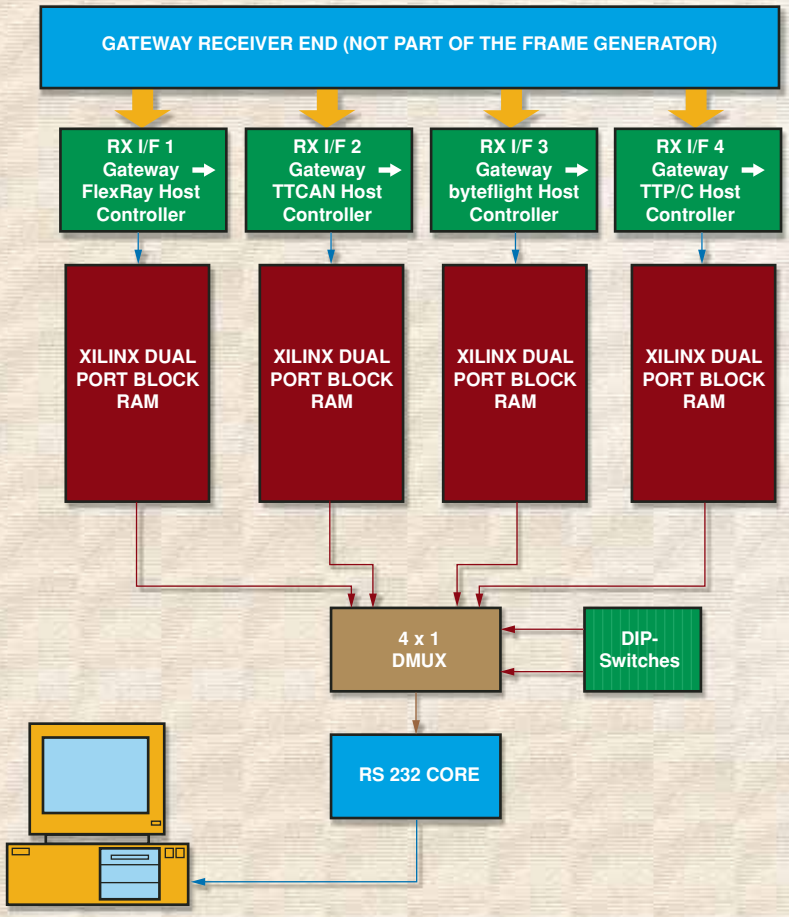


Figure 2 – Frame Analyzer building blocks

Analyzer are configured as 32 bits wide and 128 locations deep.

(Although 32-bit wide RAMs are not possible in a Spartan-II device, 16-bit dual-port RAMs can be configured as single 32-bit RAMs, as described in Xilinx application note XAPP173.)

Two DIP-switches select the RAM that the PC reads. Four switch combinations are possible – 00, 01, 10, and 11. Each combination corresponds to a particular block RAM.

The four block RAMs are read out of the FPGA sequentially and the frames are displayed on the screen.

The Frame Analyzer works in tandem with the Frame Generator, continuously transmitting 128 frames. Accordingly, the Frame Analyzer receives and stores 128 frames. The frame retrieving and displaying software then reads those 128 frames out of the FPGA.

Applications

PEI CSRC is researching the development of a real-time communication gateway for time-triggered control networks such as time-triggered controller area network (TTCAN), FlexRay™, time-triggered protocol (TTP), and byteflight™.

Some of these protocols are still under development by their respective companies and standards organizations – which means that their specifications will change. This is an ideal situation for using programmable logic for early prototyping.

The core message router of the real-time gateway is based on a round-robin packet routing mechanism. To verify this functionality in hardware, the gateway core uses programmable logic solutions from Xilinx.

The gateway's internal protocol is a proprietary protocol suitable for round-robin packet routing. The protocol conversion of TTCAN, FlexRay, TTP, and byteflight frames to gateway core format will be the responsibility of their respective host controllers.

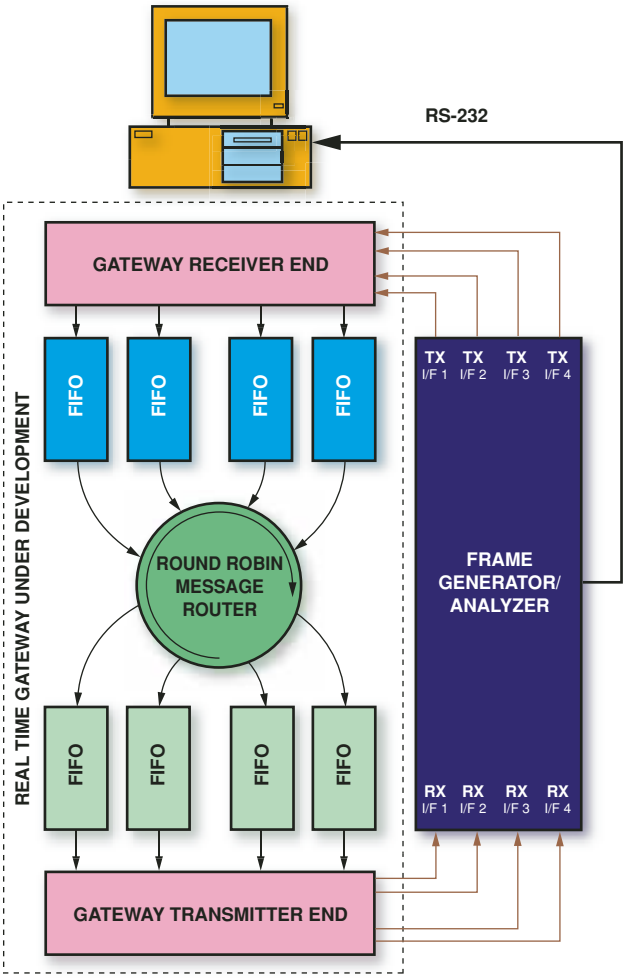


Figure 3 – Frame Generator/Analyzer application

As the Frame Generator/Analyzer emulates the interface between the gateway core and the host controllers of TTCAN, FlexRay, TTP, and byteflight, it makes the verification of the gateway core independent of the host controllers that will be developed in due course.

The Frame Generator/Analyzer can be configured for different frame rates, frame sizes, and frame sequences, allowing maximum flexibility during the testing phase to enable hard real-time testing of the gateway core. Network message schedules in time-triggered networks are defined offline and have to follow a pre-determined communication schedule. The frame sets in the Frame Generator allow for multiple combinations of frame schedules, making it possible to test the gateway's functionality and performance in a real-world sense.

The Frame Analyzer's message RAMs store all incoming frames from the gateway in the first phase of its operation. Upon reception of 128 frames, the first phase of operation is complete. The frames in individual RAMs can be now be read out through an RS-232 connection to a PC. The analysis of the retrieved frames will aid in checking and verifying the following operations of the gateway core:

- Proper Internal Frame Routing
- Data Integrity
- Non-Blocking Operation
- Dropped Frames.

Peak performance parameters of the gateway core will be verified by configuring the Frame Generator/Analyzer for different frame rates and message schedules.

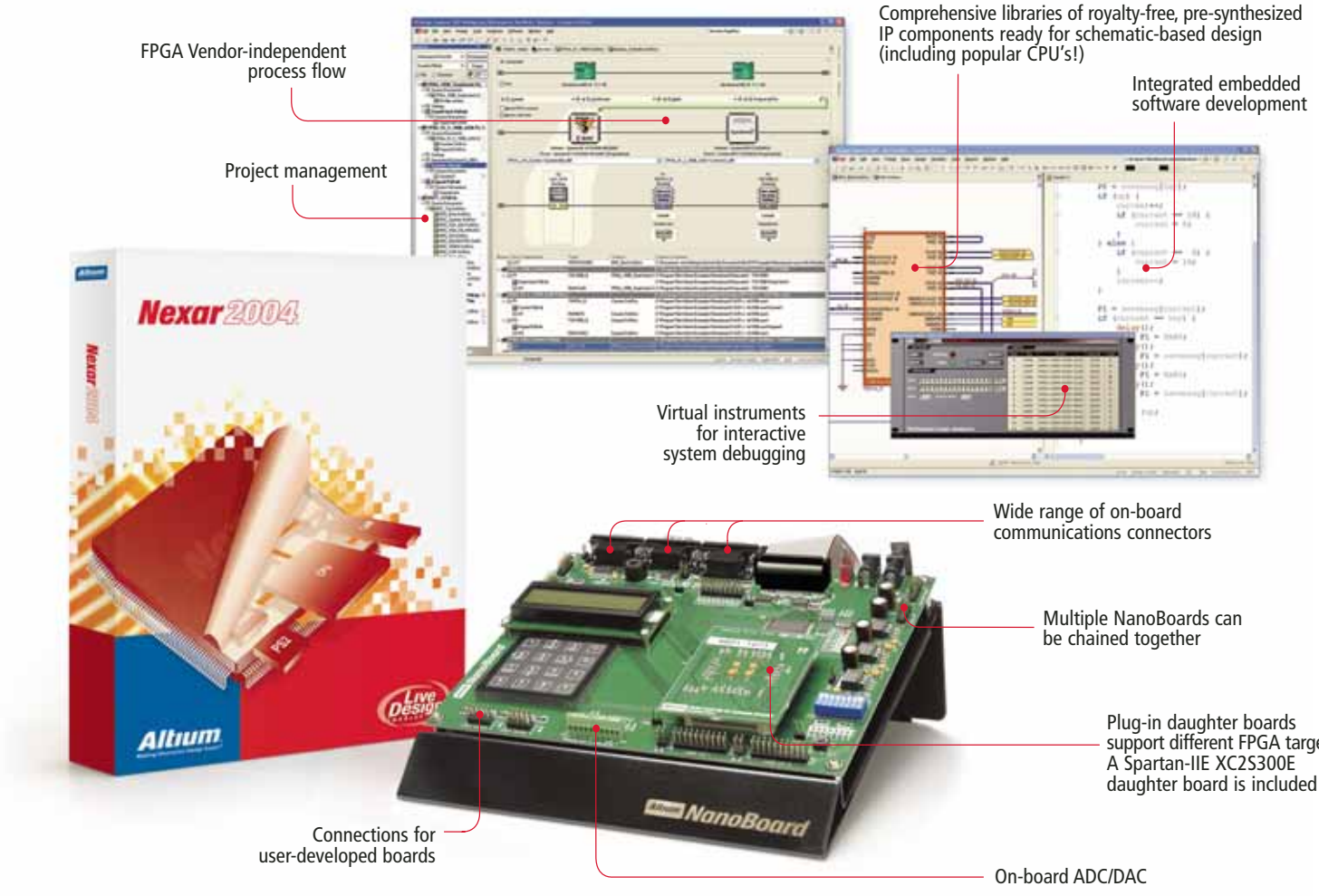
Following such an approach makes it possible to use the Frame Generator/Analyzer to test and verify the routing functionality of the gateway core early in the design phase. Figure 3 illustrates the use of the Frame Generator/Analyzer for this verification.

Conclusion

Using programmable logic solutions from Xilinx, we have been able to develop a test bed for a gateway core for time-triggered networks at a time when specifications are under review and qualified silicon is not available for all of the emerging TTPs.

The gateway core under development will be realized using Xilinx programmable logic solutions as well. As the intended application of the real-time gateway will be in automotives, the use of the IQ range of Xilinx Spartan-II devices for development and deployment will greatly reduce the cost of induction, which is one of the most critical factors in the automotive industry.

Using best-in-class, low-cost Xilinx FPGAs will enable us to deliver a real-time gateway for time-triggered networks for hard real-time applications in the automotive environment, with minimum costs incurred for design, verification, and deployment.



System-on-FPGA design made easier

Nexar is the first comprehensive development system aimed at enabling all engineers to develop complete processor based systems on an FPGA platform using methodologies analogous to board-level system design - no HDL's required!

Nexar integrates hardware design tools, embedded software development tools, royalty-free IP-based components, virtual instrumentation and a reconfigurable hardware platform to allow mainstream engineers to interactively design, debug and implement an entire embedded system inside an FPGA.

For more information on Nexar 2004, including a comprehensive multimedia demo, visit **www.altium.com/nexar** or contact your local Altium sales and support center today!

only

$7,995

Nexar includes NanoBoard hardware, software, royalty-free IP cores and much more

Nexar 2004™

Board-on-Chip development system from Altium

Altium™
Making Electronics Design Easier™

Altium Sales and Support:
Free Call: 866 764 4377
Fax: 858 521 4282
Email: nexar.sales.na@altium.com
Web: www.altium.com/nexar

Quote Value Code 1354

The Easy Path to Cost Reductions

The Virtex-II EasyPath solution offers a 99%+ fault test coverage.

by Frank Toth
Marketing Manager, EasyPath Series
Xilinx, Inc.
frank.toth@xilinx.com

A risk-free, flawless cost-reduction strategy is paramount to maintaining competitiveness in today's cost-driven environment. The Virtex-II EasyPath™ solution brings not only lower costs but also a better product, with the industry's leading test coverage.

Virtex-II EasyPath devices are identical to the corresponding FPGA. The only difference is in how they are tested – they are tested to your specific design only. Thus you get the full-feature set rather than "reduced ASIC" silicon, in a device that is fully pin-compatible and interchangeable with an FPGA.

	Virtex-II EasyPath Devices
Virtex-II Family	XC2V3000, XC2V4000, XC2V6000, XC2V8000
Virtex-II Pro Family	XC2VP30, XC2VP40, XC2VP50, XC2VP70, XC2VP100, XC2VP125,

Table 1 – Virtex Family EasyPath Devices

The Virtex-II EasyPath solution offers a 25% to 80% cost reduction with production quantity deliveries (thousands of units) in as little as 8-10 weeks (Figure 1). Device part numbers are listed in Table 1.

Identical Silicon

Conversion to a structured ASIC platform is not required. The design files used to generate a custom test program come directly from Xilinx Integrated Software Environment (ISE) software. You don't need to invest engineering resources in annotating critical paths, generating test vectors, simulation, or verification.

The silicon is identical, so prototype approval is unnecessary. You can begin high-volume manufacturing quickly, with full production quantities. Thus, you can take all the time you need to test your design and make improvements using the Virtex™-II family before committing to full-volume production using Virtex-II EasyPath devices. Only the testing is different; the silicon is identical.

Time is Money

The time gained by using the Virtex-II EasyPath solution enables you to focus valuable engineering time and resources on product feature improvements and new markets. As shown in Figure 2, a typical structured ASIC conversion requires many steps that involve direct customer engineering involvement. With EasyPath devices, no up-front ASIC software investment is required for conversion; turnaround is fast and easy because Xilinx creates the custom test program for you.

One of the limitations of the structured ASIC conversions offered by other PLD companies is the limitation of a reduced feature set when the design is converted from the prototype or initial production to high volumes. Fewer package types, I/Os, and RAM bits, as well as other features like DLLs, are available in the ASIC conversion.

Test Coverage

Virtex-II EasyPath devices rely on the same patented technology as FPGAs for testing functionality and speed, providing full test coverage at percentages of 99%+,

which is not achievable in structured ASICs (shown in Figure 3).

ASICs can typically achieve coverage between 95% and 97% using hundreds of test vectors and multiple scan chains. As ASICs grow in complexity, test issues become more acute and achieving adequate test coverage is more difficult. Iterating multiple vectors multiple times to guarantee coverage is costly and time-consuming. Plus, scan chains take silicon area and can slow the device operation.

Using multiple bitstreams, the Virtex-II EasyPath solution tests routing and logic resources with greater granularity and coverage, reaching deep inside the device to test resources that would be impossible to test using external test vectors.

This superior technology covers the corner case and virtually eliminates test escapes, which may become more prevalent as ASIC testing lags behind the complexity and density of newer, structured ASIC devices. This higher level of testing results in a better quality product shipped to customers than is available with many of today's complex ASICs.

Testing FPGAs

Virtex-II EasyPath device testing relies on the programmability and read-back capabilities of the FPGA fabric to insert control and observation points and to test the individual logic and routing resources and other complex structures, such as block RAMs and multipliers.

Used in FPGAs since their inception 20 years ago, this approach is scalable, does not require additional silicon area for scan chains, and does not penalize operating speed. Because Virtex-II EasyPath devices are tested identically to FPGAs for your specific designs, at-speed testing and logic resource and routing resource coverage are guaranteed.

A Better Test Methodology

By carefully examining each of the resources used in your design, Xilinx proprietary test generation software determines what should be tested and what combination of tests to use, as shown in Figure 4.

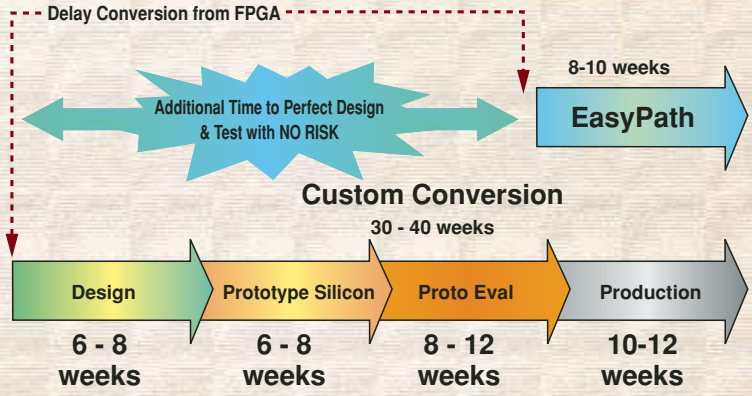


Figure 1 – EasyPath versus ASIC leadtimes

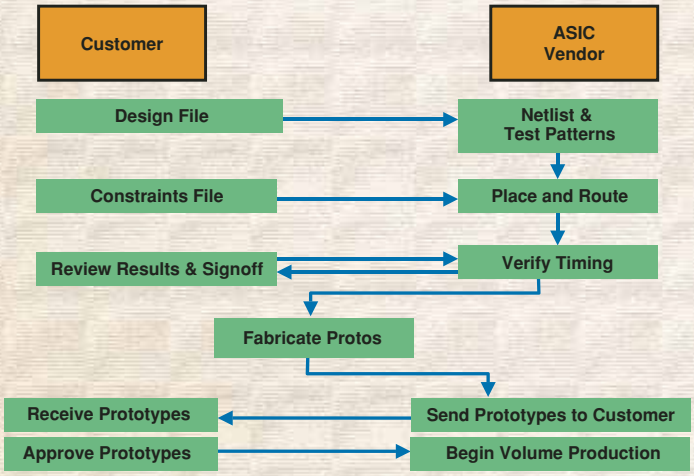


Figure 2 – Typical structured ASIC conversion flows

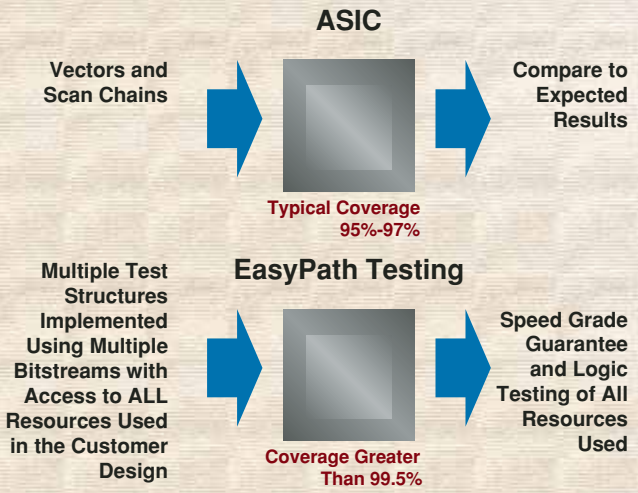


Figure 3 – Test strategy differences between EasyPath and typical ASICs

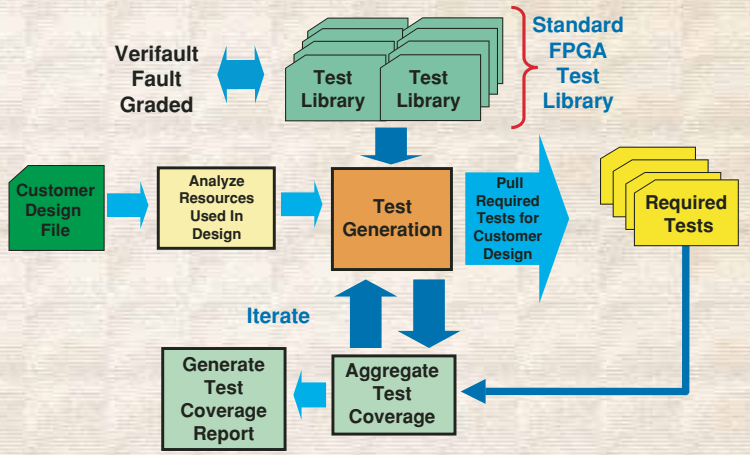


Figure 4 – EasyPath test generation flows

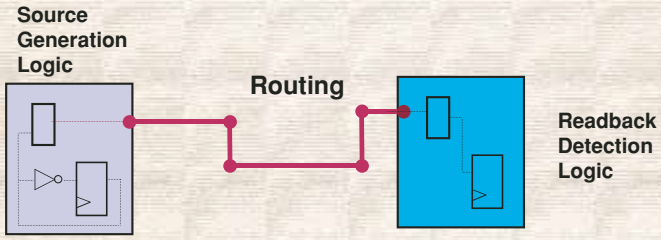


Figure 5 – Typical source/readback routing test logic

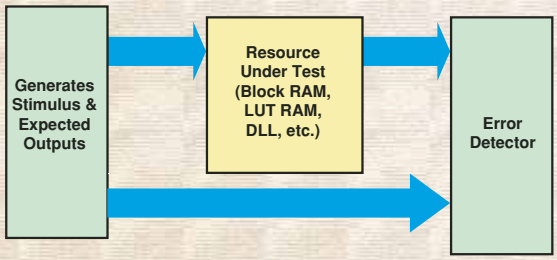


Figure 6 – Using Built-In Self Test for complex logic structures

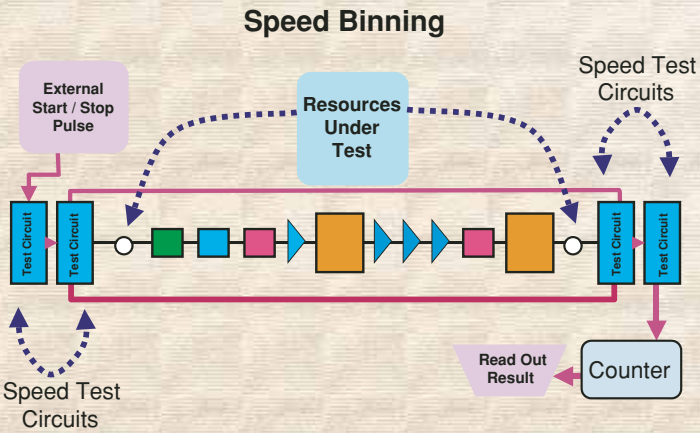


Figure 7 – Speed binning circuit for FPGAs and Virtex Series EasyPath devices

Each type of resource is measured differently. Routing resources are tested using a source/load library that toggles the selected path (Figure 5). Combinations of Built-In Self-Tests (BISTs) are used for complex logic structures. Speed grades are guaranteed by using a patented rise-and-fall time measuring method that uses a loop and counter to provide highly accurate speed grade measurements.

Testing Routing Resources and BIST

Because the FPGA fabric is reprogrammable, the test program uses a single bitstream to simultaneously test hundreds of routes. More complex circuits such as block RAMs, multipliers, and three-state buffers (TBUFs) require that BIST circuits (Figure 6) are instantiated using multiple bitstreams. These BIST circuits are based on industry-standard test structures that are identical to those used to test ASICs or any other complex logic.

Speed Grading

Xilinx assigns Virtex-II EasyPath device speed grades the same as FPGAs, as shown in Figure 7. A patented speed binning circuit tests the transition rise-and-fall times. A loop, similar in concept to a ring oscillator, outputs to a counter. Various resources under test are placed in the middle of this loop.

The number of pulses generated in a specific time is directly related to the speed grade of a Virtex-II EasyPath device. Unlike a simple ring oscillator, these circuits are capable of precisely measuring both rise and fall times to guarantee precise transition times through the circuit under test.

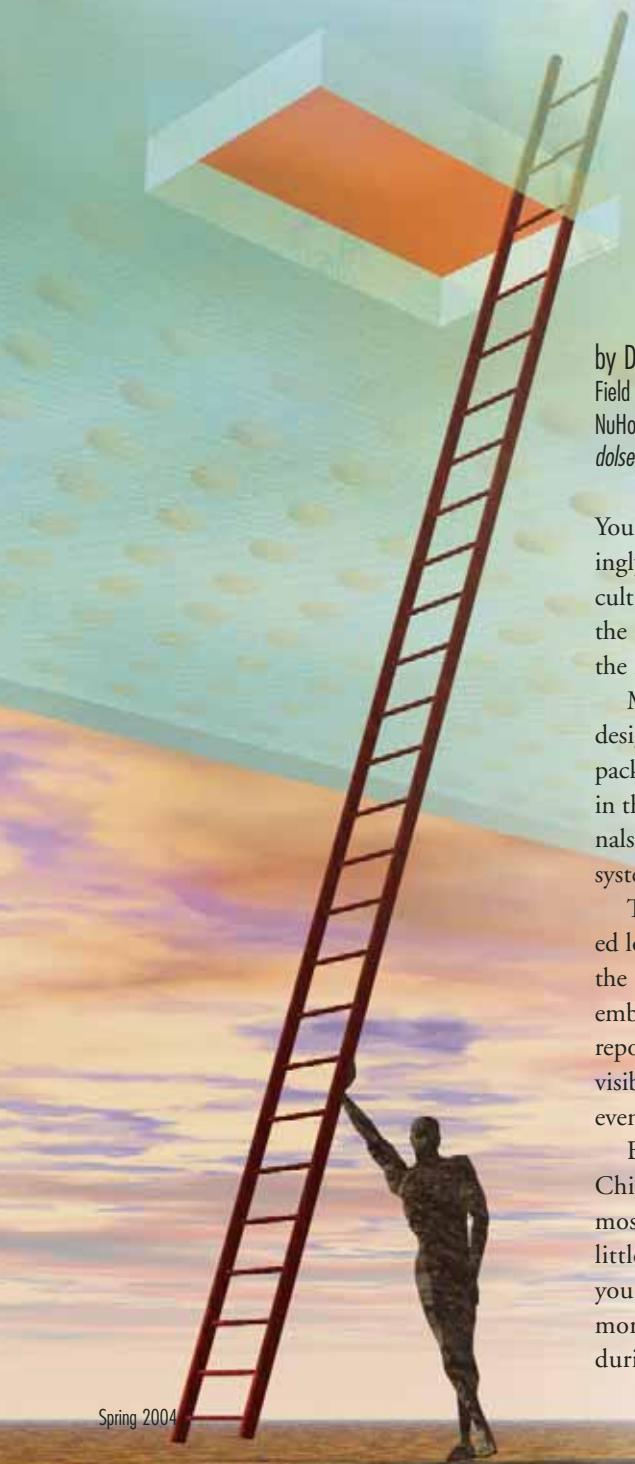
Conclusion

The Virtex-II EasyPath solution is easy, fast, and cost-effective. You can now spend more time in beta testing, ensuring that your system is bug-free and has all the required features, and swing directly into production without the need for prototype evaluations.

For more information about Virtex-II EasyPath devices, visit *www.xilinx.com/easypath.*

Think Outside the Chip

Use the Xilinx ChipScope Pro integrated logic analyzer for FPGAs as a board- and system-level diagnostic tool.



by Douglas W. Olsen
Field Applications Engineer
NuHorizons Electronics
dolsen@nuhorizons.com

Your FPGA designs have become increasingly dense and complex. They are difficult to debug because more and more of the relevant signals are buried deep within the logic fabric.

Moreover, your board and system designs are also more complex and densely packaged. Whether you're troubleshooting in the lab or in the field, your access to signals in the FPGA, on the board, or in the system is very restricted.

The Xilinx ChipScope™ Pro integrated logic analyzer (ILA) has solved much of the problem at the FPGA level. You can embed its real-time data collection and reporting cores in your FPGA to provide visibility to internal FPGA signals and events.

But if that's all you're using ChipScope for, you're not getting the most out of this powerful tool. With a little forethought during the design of your board and system, you can extract more value from the ChipScope Pro ILA during the entire product life cycle – from initial debug of board-level signals in the lab through system diagnosis and maintenance in the field.

ChipScope Pro – A Review

ChipScope Pro is a "logic analyzer in an FPGA." It provides real-time debug and verification capabilities for your FPGA design. You can find all the details about the ChipScope Pro ILA at *www.xilinx.com/ise/verification/chipscope_pro.htm*.

You interact with ChipScope Pro using a familiar logic analyzer interface running on your PC, which is connected to your hardware under test via a JTAG interface.

Figure 1 illustrates a simple ChipScope Pro design example.

ChipScope Pro Basics

The ChipScope Pro program consists of two main core types in the FPGA, plus software running on your PC:

- ILA core – With the assistance of a core inserter program, you embed the ChipScope Pro ILA core in your FPGA design to collect data when trigger conditions are satisfied. You can easily tailor the data size, target signals, and basic trigger architecture during the design phase. You can set flexible trigger conditions at debug time. (Target signals can be changed using the Xilinx FPGA editor probe tool without requiring re-synthesis, re-place, or re-route of your design.) As many as 15 ILA cores can be embedded into one FPGA.

- ICON core – You embed the integrated controller (ICON) core in your FPGA design to control as many as 15 ILA cores. This ICON core governs each ILA core and communicates with the ChipScope software running on your PC over the JTAG interface and the Xilinx Parallel IV or MultiLINX™ cable.
- ChipScope PC software – You interact with the hardware under test using ChipScope Pro software running on your PC. You spend minimal time learning to use this software, because it is so similar to a hardware logic analyzer. You can set trigger conditions and examine collected data using this software. You can also export collected data in a variety of file formats for analysis by other tools, or for incorporation into documentation.

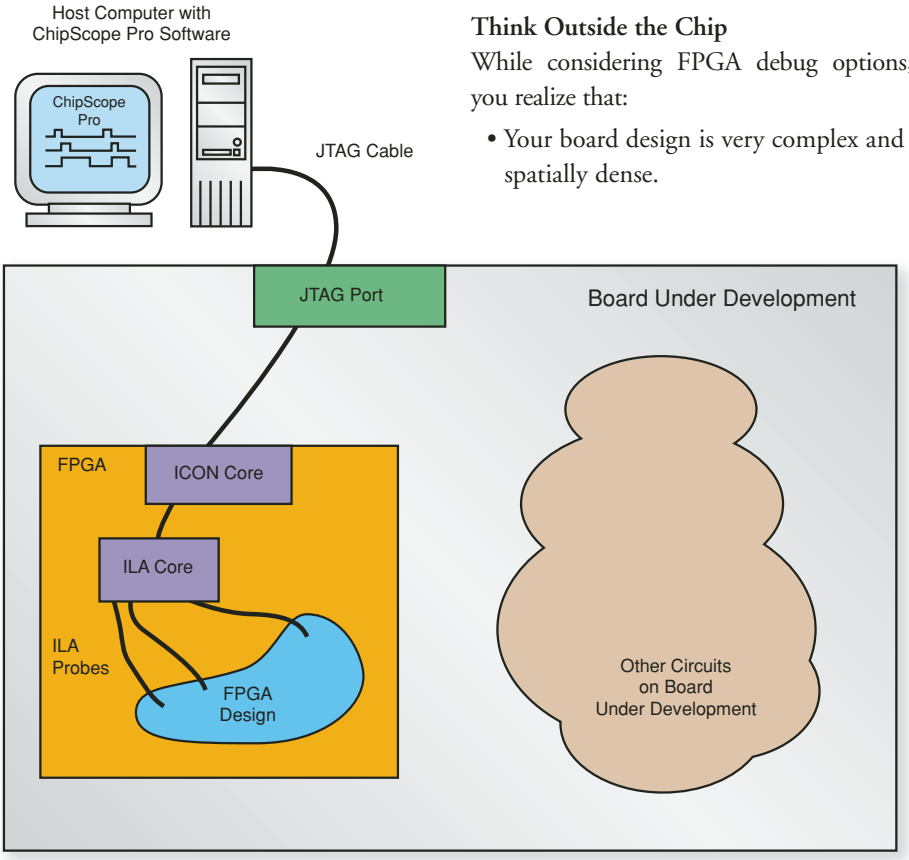


Figure 1 – A simple ChipScope Pro design example

ChipScope Pro at the Board Level

Consider this: As the lead engineer for a new board design, your job is to compose the block diagram. Your knowledge and duties include:

- Partitioning a multitude of complex functions into various circuits on the board
- Allocating many of the circuits to reside in one or more Xilinx FPGAs on the board
- Recognizing the need to provide a means of debugging the many internal FPGA designs
- Realizing that the ChipScope Pro tool can economically provide visibility to all FPGA signals
- Knowing that the ChipScope Pro tool requires no more design resources than the JTAG interface already provided for FPGA configuration
- Using some incremental FPGA resources (mostly block RAM) for ChipScope Pro data collection.

Think Outside the Chip

While considering FPGA debug options, you realize that:

- Your board design is very complex and spatially dense.
- The board circuits outside your FPGAs are not going to be any easier to debug than the internal FPGA designs.
- You do not have room to provide test point headers or connectors for all the board signals that will require further examination.

Should you route these board-level signals to spare pins on the FPGA?

Yes – then you can connect the signals to the ChipScope Pro ILA core provisioned for internal FPGA signal debugging. If you anticipate that your primary FPGA functional designs will consume all of the I/O pins or internal logic resources, you must consider upgrading to the next package size or logic density. The following benefits will outweigh the incremental cost:

- Decreased debug and development time
- Flexible, permanent debug signal routing
- Built-in board-level instrumentation, which leads to
- Reduced demand for scarce lab instruments during peak debug activity.

Figure 2 illustrates ChipScope Pro utilized as a board-level test tool. Notice its similarity to Figure 1, where ChipScope Pro is used as an FPGA-only tool.

ChipScope Pro at the System Level

You, as the system engineer for your company's sophisticated new product, must achieve multiple objectives, such as:

- Ensuring that system development and integration flow as smoothly as possible, thus minimizing your time to market
- Providing an easily maintainable product to minimize service costs for the product's overall life cycle.

Incorporate Instrumentation

What if every system that you integrate and ship could include the capability of built-in instrumentation? By utilizing the same Xilinx ChipScope Pro program that your board-level engineers use to debug FPGA designs, you can configure your products for system integration and field maintenance.

This will cost you nothing more than a few judiciously chosen system-level signals routed to a Xilinx FPGA; some incremental FPGA resources (mostly block RAM); an accessible JTAG connector; and some up-front system design time and planning. These costs will be overshadowed by the following benefits:

- Decreased system integration time
- Built-in system-level instrumentation
- Reduced demand for scarce lab instruments during peak integration activity
- Flexible, permanent maintenance and diagnostic signal routing.

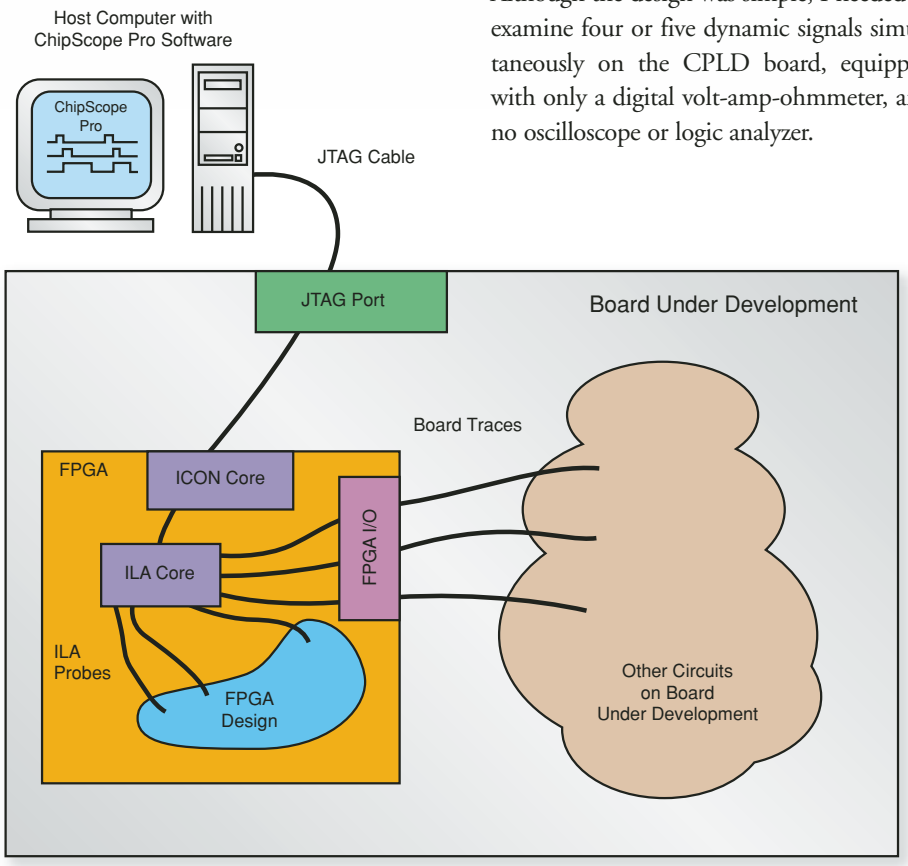


Figure 2 – ChipScope Pro board-level tool design

During system integration, engineers can connect a PC running ChipScope Pro software to the system JTAG port, and monitor system-level signals.

You don't have to use a dedicated hardware logic analyzer. Production test engineers can readily adopt this capability and use system signals during acceptance testing.

System installers can utilize ChipScope Pro's capability at customer sites to debug problems, or to ensure proper operation. Going forward, maintenance personnel can troubleshoot system problems using no more than a laptop PC running ChipScope Pro software.

Figure 3 illustrates ChipScope Pro used as a system-level test tool, which builds on the board-level test tool of Figure 2.

Design Example

I developed the concept of extending ChipScope Pro's utility when I undertook a design in my poorly equipped basement lab using a Xilinx CoolRunner-II CPLD evaluation board from NuHorizons Electronics. Although the design was simple, I needed to examine four or five dynamic signals simultaneously on the CPLD board, equipped with only a digital volt-amp-ohmmeter, and no oscilloscope or logic analyzer.

Necessity Is the Mother of Invention

Fortunately, I also had a Xilinx evaluation board containing a Virtex XCV400 FPGA. Although this was a relatively small FPGA, it was more than sufficient to house the ChipScope Pro ICON core and a simple eight-channel ILA core.

In this particular case, my implementation of ChipScope Pro was the ultimate in simplicity: no internal FPGA signals were connected to the ILA core. Rather, the eight ILA core data/trigger signals were routed directly to I/O pins, which terminated at a header on the FPGA evaluation board. I connected wires to those pins, which I then used as probes on my CPLD board.

I was able to debug my CPLD design after just a few iterations of coding, downloading, and collection of data via the ChipScope Pro software.

For the modest cost of the ChipScope Pro utility, plus a small FPGA board and JTAG cable, I discovered that you can replicate the functionality of a logic analyzer that would otherwise cost thousands of dollars.

Some Suggestions

To utilize the ChipScope Pro software as a board-level tool, follow these suggestions:

- Provide access to the FPGA's JTAG port.
- Anticipate which board-level signals you might want to examine during debug.
- Route these signals to the FPGA so you will have board-level access.
- Dedicate some extra I/O pins on your FPGA to accommodate board-level signals requiring examination.
- Anticipate some extra internal FPGA resources (mostly block RAM) for ChipScope Pro's collection of board-level signal samples.
- After analyzing the dynamic properties of the signals of interest, make sure you have an appropriate clock signal so the ChipScope Pro tool can process your board-level signals.

To utilize the ChipScope Pro program as a system-level tool, follow these suggestions:

- Provide access to an appropriate system-level JTAG port.
- Anticipate which system-level signals you will want to examine during system integration and life cycle support.

- Route these signals on boards and backplanes or cables to the FPGA chosen to host the system-level JTAG interface.
- Dedicate adequate FPGA resources (I/O pins and logic) to accommodate the system-level signals of interest.
- After analyzing the dynamic properties of the signals of interest, make sure you have an appropriate clock signal so ChipScope Pro can process your system-level signals.
- In the unlikely case that your system contains no other FPGAs, it would be worth the cost of a small FPGA in which to embed the ChipScope Pro core as an economical integrated system-level logic analyzer.

Conclusion

You can add value to your FPGA-based product by expanding the use of the Xilinx ChipScope Pro integrated logic analyzer "outside the chip" of the FPGA design.

You have already invested in ChipScope Pro for debugging the internal FPGA design. You can decrease your board- and system-level debug time for zero additional tool costs, and minimal up-front design time and resources. Additionally, you can enhance your product with permanent system diagnostic and maintenance capabilities for its entire life cycle.

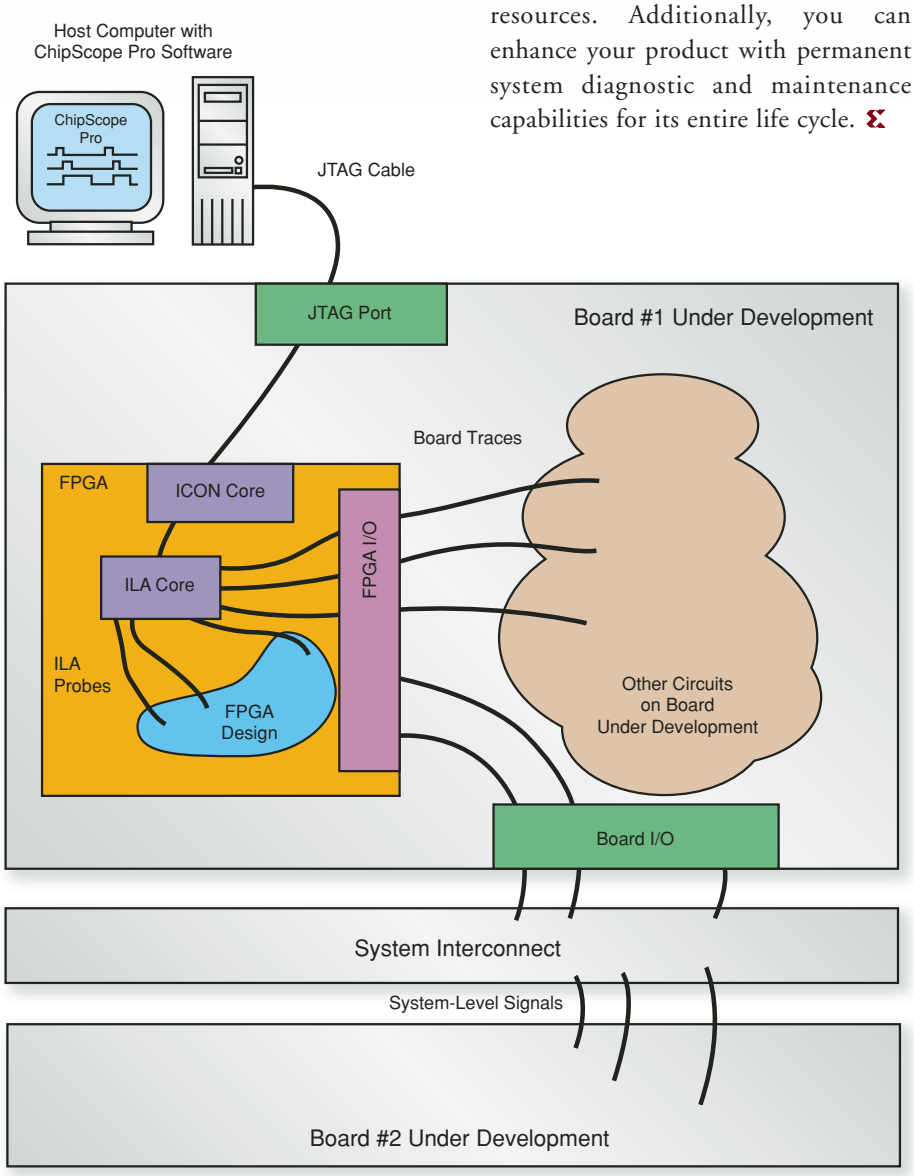


Figure 3 – ChipScope Pro system-level tool design

Two Thumbs Up

ChipScope Pro engineers endorse design methodology.

"I think it's a great idea – especially for those who don't need or can't afford some of the higher-end features provided by other debug solutions, like timing mode triggering or deep trace memory," said Brad Fross, design engineering manager for Xilinx Advanced Products Division Systems Engineering.

"This is a new, cost-effective solution that expands the practical application of ChipScope Pro tools beyond the chip," agreed Brent Przybus, product marketing manager for the Advanced Products Division.

"The only tricky part of Doug Olsen's solution will be clocking," Przybus noted, "but I believe Doug covers this adequately in his article, given that each system will be slightly different."

Fross elaborated on this caution: "On-chip debug is pretty straightforward, because the ILA core is taken into consideration during timing-driven design implementation. However, it definitely gets trickier to do when off-chip signal delay and external clock synchronization need to be taken into account.

"In any case, I do think this is a great article to get people thinking about how to debug their systems way before they actually find themselves needing to do so," Fross concluded.

ChipScope Pro

ThumbPod Puts Security Under Your Thumb

UCLA students develop a prototype fingerprint authentication device incorporating a Virtex-II FPGA.

by Patrick Schaumont
Ph.D. Candidate, Embedded Security,
EE Department, UCLA
schaum@ee.ucla.edu

Ingrid Verbauwhede
Associate Professor, Embedded Security,
EE Department, UCLA
ingrid@ee.ucla.edu

We live in a networked world, where your digital alter ego is almost as important as your physical self. Your digital identity, usually comprising a series of electronic records, determines whether you can buy a car, get a loan, or open a bank account.

Yet the link between your physical and digital selves is surprisingly weak and insecure. Identity theft and credit card fraud are ubiquitous. Indeed, we have grown used to verification via trivial secrets such as your mother's maiden name. But trivial secrets require only trivial guesswork to be revealed.

A more unique identifier is possible through the use of biometrics – unique physical features such as DNA, fingerprint data, iris composition, or facial structure. These are distinctive keys that cannot be forgotten or lost and, with appropriate precautions, are difficult to counterfeit.

The ability to authenticate yourself through biometrics opens up enormous possibilities in embedded and portable contexts, from key chains to credit cards. The extraction of unique features from biometrics data, however, is a complex signal-processing problem that requires a significant amount of computational power.

One solution may be ThumbPod, an FPGA-based prototype (Figure 1) of an embedded fingerprint authentication system created by seven graduate students and their advisor at the University of California, Los Angeles. ThumbPod (*www.thumbpod.com*) includes a fingerprint sensor, an embedded processor with memory, dedicated processing units, and a serial communication channel.

According to design specifications, the unit must deliver 1,000 fingerprint-matching operations, each comprising an estimated 250 million operations, on a single battery. Given the complexity of the embedded fingerprint recognition, we use an FPGA prototype to quickly and easily evaluate critical design decisions.

In its final form, ThumbPod could be integrated into an embedded device not larger than a key chain fob (Figure 2).

The Security Pyramid

A security application is only as safe as its weakest link. ThumbPod is no exception. Security must be considered at all design stages and design abstraction levels.

We employ a "security pyramid" to identify four different design abstraction levels in ThumbPod that affect safe operation:

- The *protocol* level expresses how ThumbPod connects to an application; that is, how it will be used to open an electronic lock. A critical aspect in the design of this protocol is that fingerprints are sensitive personal data, and thus should be processed with adequate privacy. Unlike many other biometrics systems, our ThumbPod processes all fingerprint data locally. Even the fingerprint sensor is local and private for each person, and cannot be shared.

- The *algorithm* level collects the functions and algorithms used as building blocks in the ThumbPod security protocol. This includes fingerprint signal processing to extract a unique template from a fingerprint, consisting of minutiae. It also includes encryption using the Advanced Encryption Standard.

- The *architecture* level defines the target onto which the algorithms are mapped. We combine a soft-core, 32-bit,

Figure 1 – ThumbPod prototype

LEON2 SPARC processor with hardware acceleration co-processors for signal processing and encryption. This enables the general-purpose processing required to integrate the application, as well as the intensive data processing necessary for fingerprint matching, to operate in an embedded context.

- The *circuit* level deserves special attention, because it is a potential backdoor for security hackers. With careful technology mapping, however, we can make ThumbPod resistant to power and timing attacks.

Figure 2 – ThumbPod conceptual drawing

System Architecture

ThumbPod's high design complexity makes it impossible to capture everything in one design layer. Instead, we approached the design as a stack of three machines, each one bootstrapping the next (Figure 3).

At the bottom layer is a Virtex™-II XC2V1000 FPGA. It has a rich set of on-chip resources providing interconnection, on-chip storage, and logic. The prototype is implemented on an Insight Electronics development board that also offers 32 MB of DDR RAM for background storage.

On the FPGA, we configured a soft-core processor with two hardware co-processors. The processor is a SPARC-compliant, 32-bit LEON2 from Gaisler Research (*www.gaisler.com*). An encryption processor and a discrete Fourier transform (DFT) signal processor provide acceleration for the critical processing parts of the design.

On top of the LEON2, we ran an embedded Java kilobyte virtual machine (KVM). The KVM offers our application developers a smooth transition from application development to the embedded design.

In addition, all platform-specific features, such as the fingerprint sensor or hardware co-processors, can be integrated into a single programming environment using native interfaces. (The ThumbPod website [*www.thumbpod.com*] collects additional publications that elaborate on the performance and test issues occurring through the use of these interfaces.)

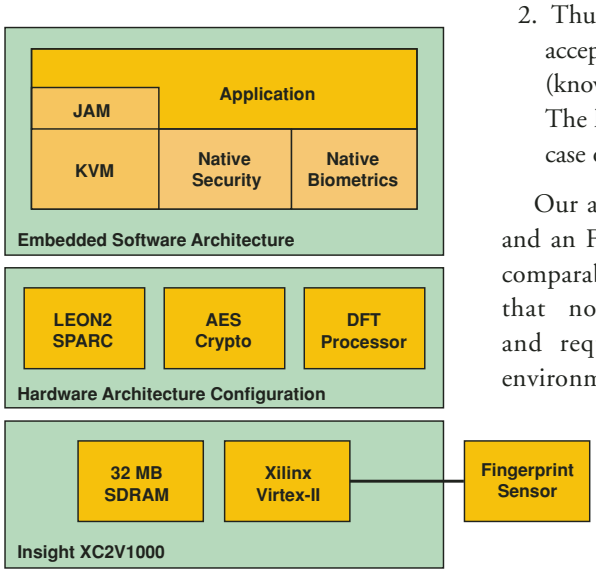


Figure 3 – ThumbPod system architecture

Fingerprint Matching

When you put your finger on the fingerprint sensor, it takes an image. From this raw image, we extract a set of unique features, known as minutiae (Figure 4) through a sequence of image processing steps (Figure 5). We obtain the minutiae beginning with the raw grayscale image of the fingerprint and proceeding through a sequence of image processing steps.

The set of minutiae forms a secret key that is stored securely in ThumbPod as a template. It cannot be changed afterwards, nor can it ever be extracted. During actual use, this template is matched to a newly captured fingerprint. This matching process returns a score between 0 and 100, which is used to decide acceptance or rejection of the ThumbPod user.

The bulk of the processing power in ThumbPod is devoted to the extraction of minutiae. We started with a reference software implementation from NIST (National Institute of Standards and Technology). Subsequently, we adapted this for embedded processing. This includes conversion from floating-point to fixed-point operation, reduction of the memory requirements, and the introduction of a DFT hardware accelerator.

The operation of ThumbPod must be as reliable as possible. We verified the quality of the fingerprint detection and matching algorithms by evaluating their probability for error.

Two types of mistakes could occur:

1. ThumbPod could reject a fingerprint corresponding to the true template (known as a "false reject," or FFR).
2. ThumbPod, however, might also accept an incorrect fingerprint (known as a "false accept," or FAR). The latter case is, of course, the worse case of the two.

Our algorithms gave an FRR of 0.5% and an FAR of 0.01%. These figures are comparable to commercial-grade systems that normally run on workstations and require power-hungry processing environments.

Design Flow Using GEZEL

ThumbPod is programmed in a combination of Java, C, and VHDL. Our design flow ensures that the appropriate verifications are complete before the design is brought to the FPGA (Figure 6).

A key element of the design flow is the GEZEL design environment (*www.ee.ucla.edu/~schaum/gezel*) that supports the development and prototyping of our coprocessors. GEZEL provides instruction-set co-simulation as well as VHDL code generation. In combination with the FPGA, GEZEL technology provides a high-level exploration environment for complex system-on-chip architectures such as ThumbPod.

Our design flow consists of three distinct steps, each with an increasing amount of implementation detail.

Figure 4a – Raw fingerprint data from sensor

Figure 4b – Minutiae location with superimposed cleaned-up fingerprint image

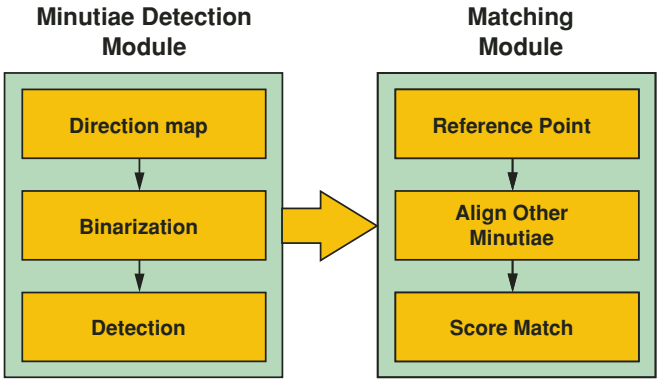


Figure 5 – Fingerprint minutiae detection and matching

We developed the top-level security protocol in Java. Using simulation on a workstation, we verified how the protocol behaves under security exceptions such as replay attacks or the use of false fingerprints. We also checked functional aspects, such as protocol communication timeouts and transmission errors.

Next, we integrated the C code of the fingerprint detection and matching algorithms into the Java code. We used the native method capabilities of Java. We developed this C code as a separate software IP core. We tested it extensively in overnight simulations before integrating it. These overnight simulations were required to check the quality loss resulting from fixed-point refinement of the fingerprint algorithms.

We then ported the ensemble of Java and C code to the embedded Java KVM. The KVM is executed on top of the LEON2 instruction-set simulator. Using this setup, we could simulate how the ThumbPod application downloaded as an applet to the prototype platform – and how the protocol worked in combination with a server application.

Once the simulation ran on top of an instruction-set simulator, we began including models of the actual target architecture, particularly the hardware coprocessors. The GEZEL environment captured cycle-true descriptions of the coprocessor architectures using a specialized programming language. The models of the AES and DFT coprocessors were co-simulated with the LEON2 instruction set simulator.

Through VHDL code generation, we obtained seamless connection to the FPGA platform. We followed a standard FPGA design flow based on Synplicity® synthesis software and Xilinx ISE software.

Conclusion

We demonstrated our ThumbPod prototype in a University Booth at the 2003 Design Automation Conference in Los Angeles. ThumbPod is a first-of-its-kind device that demonstrates portable and embedded biometrics processing as an FPGA prototype.

Embedded biometrics is a field in full expansion. As information technology further penetrates our everyday life, the need for privacy and authentication will continue to grow. Passwords and secret questions will become impractical compared to the convenience of biometrics.

But the complex signal processing that comes with embedded biometrics will require innovative architectures. These architectures combine sequential processing with the power efficiency of parallel hardware.

Using platform FPGAs and reconfigurable technology, we can build and test embedded biometrics solutions like ThumbPod in a way that outperforms the design time and cost of other technologies, such as COTS solutions or ASICs.

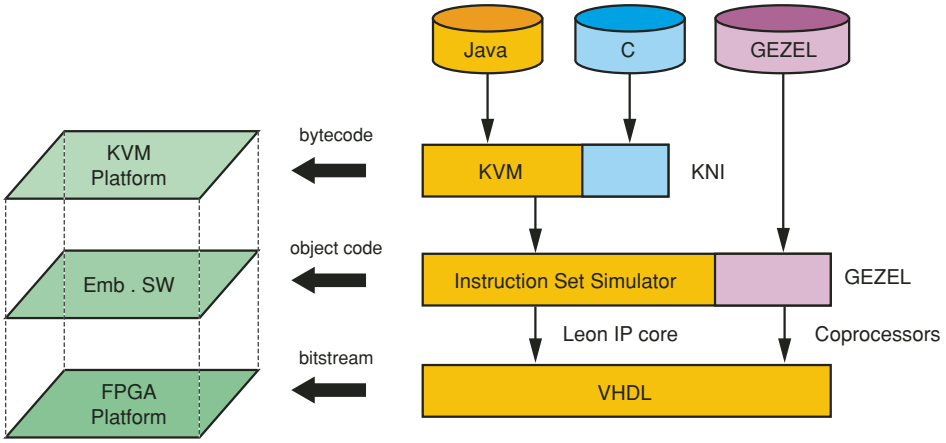


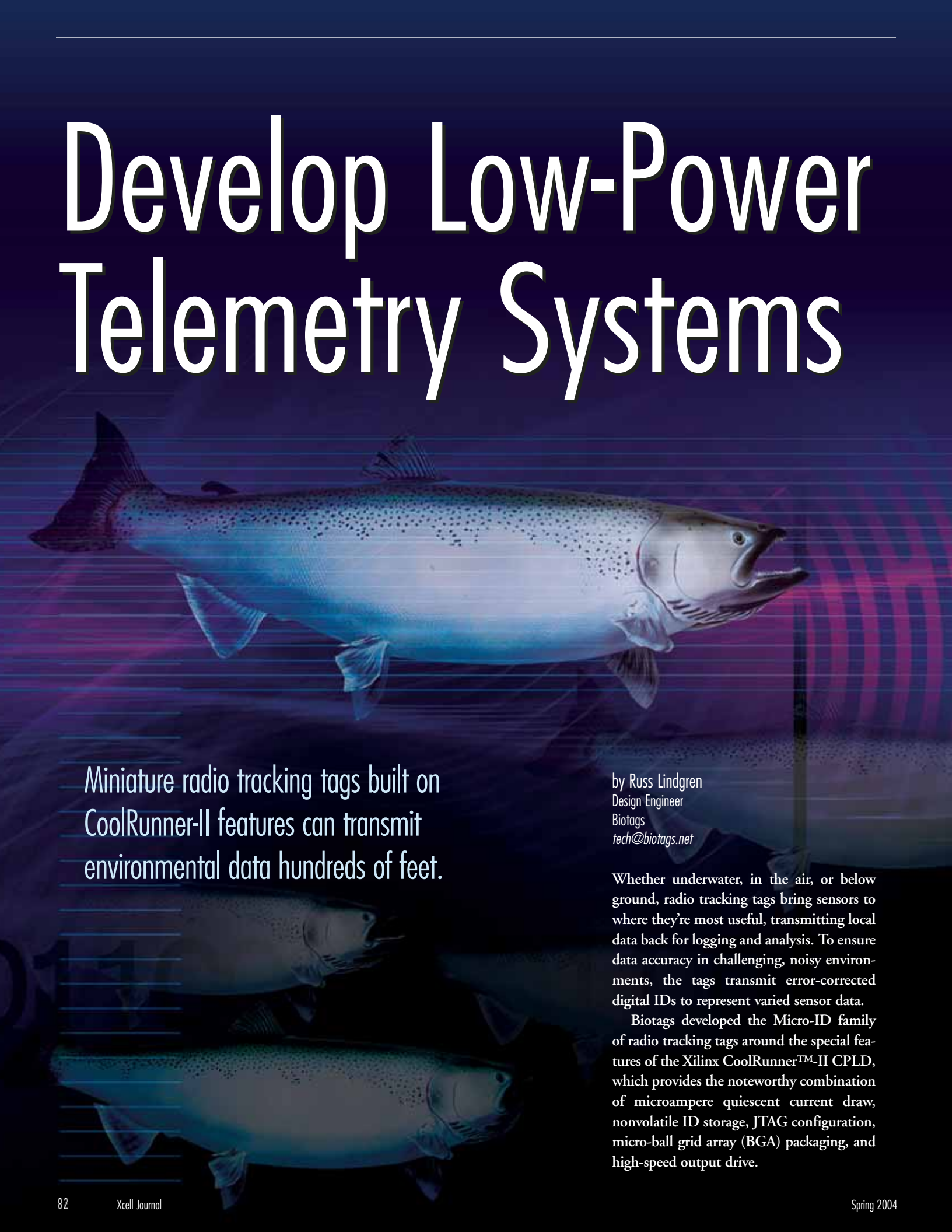
Figure 6 – ThumbPod design flow

Constructing the ThumbPod system required a significant amount of dedication and time. ThumbPod is the result of the efforts of seven graduate students at the University of California, Los Angeles, over a period of eight months. They are Yi Fan, Alireza Hodjat, David Hwang, Bo-Cheng Lai, Kazua Sakiyama, Patrick Schaumont, and Shenglin Yang.

The project enjoyed the support of many people, including the students' advisor, Professor Ingrid Verbauwhede, Jiri Gaisler from Gaisler Research, and Daryl Specter from Insight Electronics.

We would also like to thank the Xilinx University Program team, including Jeff Weintraub and Anna Acevedo, for their support and generous donation.

Develop Low-Power Telemetry Systems



Miniature radio tracking tags built on CoolRunner-II features can transmit environmental data hundreds of feet.

by Russ Lindgren
Design Engineer
Biotags
tech@biotags.net

Whether underwater, in the air, or below ground, radio tracking tags bring sensors to where they're most useful, transmitting local data back for logging and analysis. To ensure data accuracy in challenging, noisy environments, the tags transmit error-corrected digital IDs to represent varied sensor data.

Biotags developed the Micro-ID family of radio tracking tags around the special features of the Xilinx CoolRunner™-II CPLD, which provides the noteworthy combination of microampere quiescent current draw, nonvolatile ID storage, JTAG configuration, micro-ball grid array (BGA) packaging, and high-speed output drive.

In addition to the 1.8V CoolRunner-II device, Biotags' Micro-ID core functions require just a watch-crystal oscillator for time base and a gated-crystal oscillator for radio frequency (RF) carrier generation – an integration that results in a functional single board data transmitter contained in a 0.8-cm square (8 mm by 10 mm).

Customizing Micro-ID tags to different applications often requires only source-code changes to the CPLD programming, providing a significant cost advantage over ASIC-based radio tags.

Radio-Tagged Salmon

In fisheries research and population studies, biologists often trace downstream salmon migration by placing radio tracking tags in the smolts and tracking the tagged fish as they pass around dams. Because salmon smolts are typically under 20 cm long, they require a tiny radio tag. As shown in Figure 1, the smallest version of the Micro-ID radio tag measures 27 mm long and 10 mm wide, and has a mass of less than 2.5 grams.

To monitor migrations from river to ocean, researchers set up antennas and receivers at crucial areas around dams where salmon and predators are most likely to meet. Typically, the receivers can detect a tag at a distance exceeding 1,000 feet, even when the tag is at a water depth of 6 feet. Because each tag has its own set of custom IDs, researchers can track the route of each individual salmon through miles of river.

Location tracking is only part of the story, however. Sensor data provides far more specific information on survivability. For this application, the sealed Micro-ID tag continuously monitors salmon body temperature for a period of three days. If the temperature exceeds 80°F (the digital switch point), then the salmon didn't survive and likely served as prey for a mammal or bird.

To differentiate these two environmental states, the Micro-ID tag stores the change and transmits a different ID after the temperature switch point is triggered, allowing researchers to review the transition moment in the data log. In this way, radio tracking tags monitor and respond to their environment.

Beyond underwater radio transmission applications, the Micro-ID radio tracking tag can transmit through building materials and masonry, and thus track pipes in buildings or even under a street. In another industrial application, placing a Micro-ID tag within raw materials aids in the continuous identification of the source and movement of those materials.

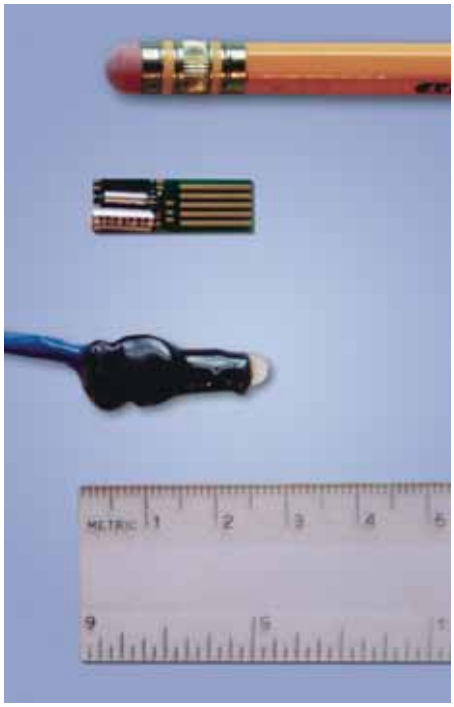


Figure 1 – Two steps to building a micropowered active RF tag: program and encapsulate. The edge connector on the Micro-ID circuit board allows configuration of each tag with unique ID codes. Once housed in waterproof epoxy and sporting an internal battery, the Micro-ID is ready for use.

Extend Battery Life

Because Micro-ID tags include a rechargeable battery, they're reusable. But when an RF ID tag includes a battery, it's defined as an active RF tag. Let's consider how the brief RF transmission time of the tag programming extends battery life.

Low-power active tag operation depends on a few crucial specifications of the CoolRunner-II CPLD: low quiescent power draw and low I cc vs. frequency. Tracking tags typically have a tiny duty cycle (a ratio of 1:32 or less) so that the higher current draw of the RF transmission portion of the cycle results in low average power consumption.

By driving the CoolRunner-II CPLD during the tag's quiet period with a low frequency clock and minimizing the timing logic, it is quite easy for a programmed 64-macrocell CoolRunner-II device to draw under 40 µA in this application.

With a judicious review of the I/O design as well as careful mapping of I/Os in Xilinx WebPACK™ software tools, you can optimize power consumption to microampere levels, allowing low-mass watch batteries to power the CoolRunner-II CPLD for days.

WebPACK Software Compacts ID and ECC Data

Although the 56-pad micro-BGA package for CoolRunner-II CPLDs contains at most 64 macrocells, you can implement at least four codes, 32 bits in length, and still have room for control logic, because WebPACK software compacts the 128 bits of serial ID data during logic reduction.

Let's consider how to build custom data codes for ID transmission at the bit level. I often construct 32-bit data codes to include synchronization (occupying the first four bits) in the error correcting code (ECC) calculation. After the synchronization bits come the ID code for the tag (14 bits), followed by 14 bits of ECC data.

The large number of ECC bits in this application provides for excellent data reliability. The type of error correcting codes and modulation techniques depends on the tag application.

Most importantly, ECCs and modulation techniques are defined in the source file for CPLD programming. With the serial ID data defined, the next step is to transmit the data via an RF carrier.

Flexible RF Modulation

Biotags' Micro-ID tags implement a data modulation scheme that makes the most of its dual-oscillator setup: a low-frequency 32 KHz watch-crystal oscillator to provide a reliable time base over a wide temperature range and a higher-frequency crystal oscillator that forms the basis for harmonic generation of the RF carrier.

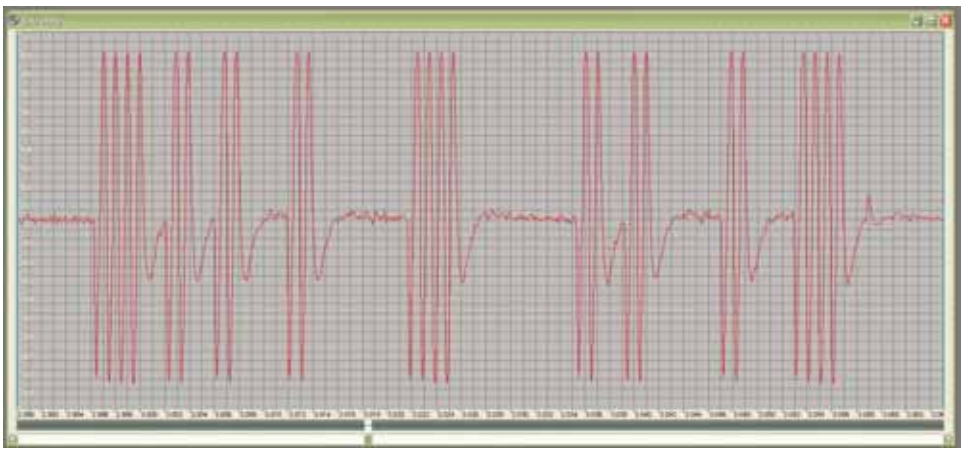


Figure 2 – Receiving data from an active radio tag is as simple as recording audio. To display the audio data stream, apply a Windows-based viewer to plot the waveform. This picture of the audio data stream, representing ID code 536, illustrates the amplitude modulation used in Biotags' salmon tracking application.

Because of the variety of wireless data transmission methods possible with this two-oscillator design, a programmable logic-based RF tag provides an excellent platform for the development of sensor and ranging applications. By using a CPLD, you can easily reconfigure the dual-oscillator setup to support many modulations, including AM, FM, and sideband. Given that you have serial bitstreams to transmit, let's consider how to implement each type of modulation.

The simplest technique is amplitude modulation (AM), where the base frequency for the RF carrier, typically in the 10 MHz-60 MHz range, is gated by the data stream. For the period of time when the tag is transmitting a "1," the CPLD enables the RF base frequency output on one or more of the CPLD's pins, generating the RF carrier via a tuned circuit. When the tag is transmitting a "0," no RF carrier is transmitted.

A popular variation of the basic AM technique provides for self-clocking of the data, which adds 1-0-1-0 transitions at twice the data clock frequency when transmitting a "1." To decode the self-clocking data, I use a known data frequency and check the ECCs. This type of data code is displayed in Figure 2.

Now that we've considered AM, digital frequency modulation (FM) is easiest to comprehend if you recall a basic premise of Fourier analysis: when modulating a high-frequency carrier with a low-frequency signal (for example, 100 MHz and a 20 KHz signal), the harmonic content of the signal shifts. It no longer includes the 100 MHz carrier. Instead, the harmonic content of the carrier contains the sum and difference of the two frequencies: 99.98 MHz and 100.02 MHz. An RF spectrum analyzer will show these two peaks as the lower and upper sideband.

To receive FM-encoded data, assume 20 KHz below the RF carrier to represent a "0"; the RF carrier represents a "1." Then tune the FM receiver midway between these two frequency ranges – 99.99 MHz – and specify a 10 KHz FM modulation.

Because multiple tags usually transmit on the same RF carrier frequency, RF carrier generation requires the use of a stand-alone gated-crystal oscillator – gated to minimize power draw during the tag's quiescent phase.

Finally, as shown in Figure 3, a circuit's physical size influences RF carrier generation, where the small physical size and minimal circuit board parasitics of the 56-pad micro-BGA package excels in radio spectrum accuracy.

Conclusion

Biological research with small juveniles of a species requires low-impact, self-powered, implantable packages with exceptional reliability. The reconfigurable featues of the CoolRunner-II CPLD enable 100% testing of each Micro-ID radio tag, an essential requirement for wild release population studies of endangered species.

Customized through CPLD programming for each biological application and local environment, Biotags' Micro-ID integrates advanced packaging, low-power operation, and wireless telemetry into a tiny battery-powered unit. One just might be monitoring a salmon swimming in a river near you.

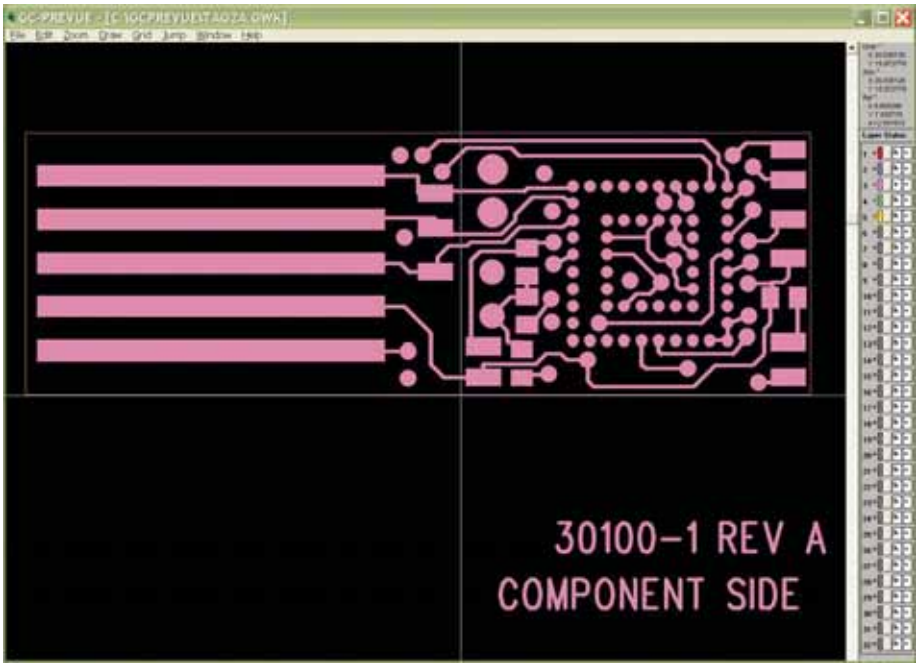


Figure 3 – The CoolRunner-II micro-BGA package is ideal for RF carrier generation because it reduces circuit board parasitics. The removable edge connector on the left enables programming and testing before final radio tag assembly.

Xilinx Teams with Optos

Xilinx Design Services assists in the development of a new eye care technology.

by Barrie Mullins
Integrated Solutions Manager, Xilinx Design Services
Xilinx, Inc.
barrie.mullins@xilinx.com

Optos PLC, a revolutionary eye care technology company, selected Xilinx Virtex-II Pro™ FPGAs to be at the heart of their wide-field visual imaging system. To lower the risk, reduce the learning curve, and meet their time-to-market requirements for this new technology, Optos engaged Xilinx Design Services (XDS) to help them plan, design, implement, and deliver their complex solution.

The Optos Panoramic System comprises two IBM™ PowerPC™ 405 microprocessors and multiple proprietary interfaces, as well as industry standard interfaces. The design utilizes the latest technology, tools, and software provided by Xilinx and gave XDS the opportunity to apply its system, hardware, and embedded software knowledge, along with its project management expertise, to meet the time-to-market requirements.

The Design Challenge

For Optos, the challenge was to find a technology that provided them with the technical features to take their system from concept to reality. The system is used by a qualified operator to take a scan of the patient's eye while they look at a target presented on an LCD panel. This image capture uses Optos' patented technology in wide-field visual imaging and transfers the data to a microprocessor. The processor is used for storage of patient data and processing by the operator.

The challenge for XDS was to design, develop, test, simulate, integrate, and bring up the hardware and software on a single platform to enable a complete connectivity solution addressing all layers of the Optos application. The solution also had to be scaleable for future features and upgrades. XDS was also required to meet budget constraints.

The Teams

The Xilinx Design Services team worked with the Optos engineering team to set the system requirements and to agree on a schedule and deliverables for the project. Over the period of the project, the XDS project manager worked with the Optos project management team to ensure that milestones were delivered and that information flowed smoothly between the two teams.

Optos

Optos was founded to develop a technology that will provide ophthalmologists and optometrists with improved image capture and analysis capability for the early detection and prevention of eye disease. Optos' patented technology in wide-field visual imaging is unique. Their business strategy is to make their visual imaging system available to the greatest number of practitioners, thus allowing an overall improvement in eye care and early disease detection for a far greater number of people.

The Optos headquarters in Dunfermline, United Kingdom, is the base for research and development of the Panoramic diagnostic ophthalmic apparatus, the technology behind Optomap Retinal Images.

Xilinx Design Services

Optos contracted XDS to design and deliver the Virtex-II Pro 2VP20 design. By doing so, they reduced the risk and learning curve for this new technology, and they ensured they would meet their time-to-market goal. XDS gathered a team of experienced co-design engineers with in-depth knowledge of project management, embedded software

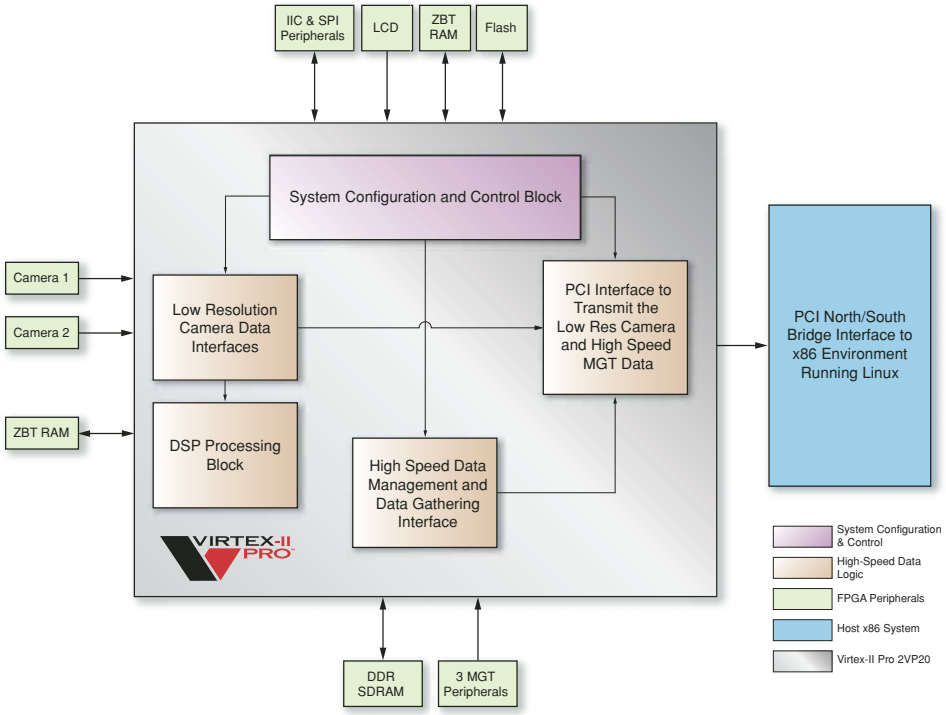


Figure 1 – Block diagram of Virtex-II Pro design in the Panoramic system

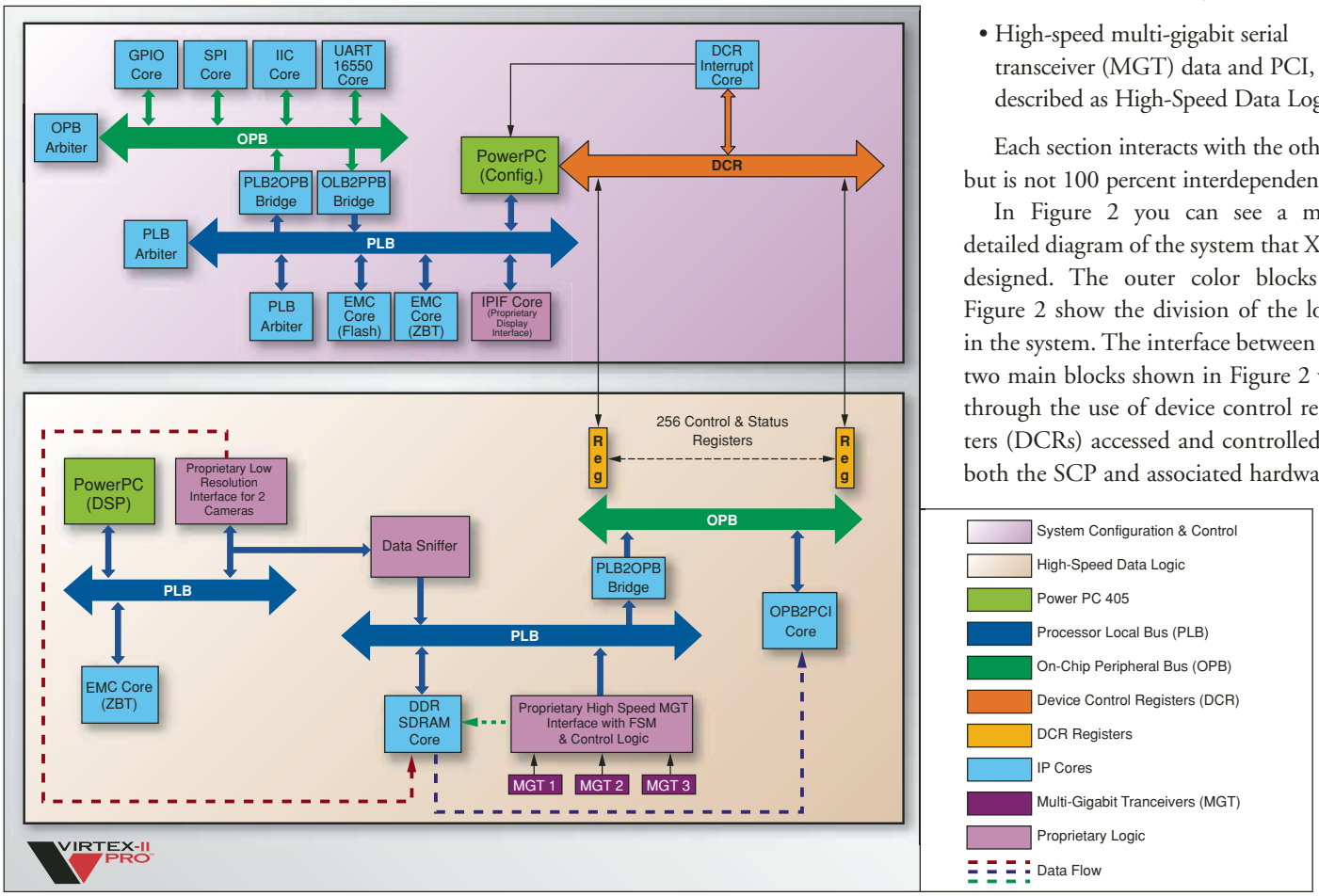


Figure 2 – Detailed view of Virtex-II Pro design in the Panoramic system

design, FPGA design, co-design tools, IP cores, and platform FPGAs.

The final delivery to Optos included the source code for the embedded test software, HDL designs, design scripts, HDL test vectors, simulation test software, build scripts, and associated testbenches. The delivery specifications also included all associated project documentation: project plan, design specifications, testbench strategy, software specifications, memory map, and verification plan.

The Panoramic System

Figure 1 shows the basic Panoramic system design, which includes:

- System control processor (SCP), described as System Configuration and Control in the diagram legend
- Camera and digital signal processing (DSP) components, described as DSP and Camera Logic
- High-speed multi-gigabit serial transceiver (MGT) data and PCI, described as High-Speed Data Logic.

Each section interacts with the others, but is not 100 percent interdependent.

In Figure 2 you can see a more detailed diagram of the system that XDS designed. The outer color blocks in Figure 2 show the division of the logic in the system. The interface between the two main blocks shown in Figure 2 was through the use of device control registers (DCRs) accessed and controlled by both the SCP and associated hardware.

System Control Processor

The main function of the SCP is system configuration management, coordination, status reporting, and control of the timing of events based on inputs. It starts image capture, and controls LCD display and other logic through registers in the system.

The main technical blocks of the SCP perform the following functions:

- Communicate with the Intel™ x86 operating system environment over PCI and RS-232 connections
- Configure the LCD panel and camera interfaces via IIC
- Display images on the LCD panel
- Communicate with peripheral subsystems via SPI
- Boot from internal Virtex-II Pro block RAM and load application from external flash memory to ZBT RAM before running it
- Schedule and control events in the application software via interrupts and GPIO
- Use DCR registers to gather status and control operations in other sections.

DSP and Camera Logic

The camera interfaces are required to support 30 frames per second (fps) of 640 x 480 pixels, requiring bandwidths greater than 9 Mbps for each camera. The data is stored locally in ZBT RAM so that a DSP algorithm running on the second PowerPC 405 microprocessor can be applied to the data. While the data is being received from the cameras, it is also copied to DDR SDRAM for transmission over the PCI to the host x86 memory space.

High-Speed Data Logic

RocketIO™ MGTs are used to receive high-speed data from three peripheral boards, each transmitting data from a single MGT on a Virtex-II Pro 2VP7. The MGT data transfers are initiated by an external system triggered by the operator. This event is synchronized using interrupts to the SCP. The data from the three MGTs is formatted in the Virtex-II Pro device and used to form a larger data packet, which is then stored in DDR SDRAM for transmission over PCI.

The PCI section is used to transfer all the data stored in the DDR SDRAM to the x86 host processor environment. XDS designed a number of proprietary hardware blocks to control the flow of data from DDR SDRAM to the OPB-PCI core and over to the host x86 memory space. Configuration and status of the PCI core is carried out by the SCP through DCR registers.

Tools

During this project, XDS used several software tools to create the Panoramic system, to simulate the design, and to manage the development environment to ensure traceability and quality releases. The tools used are listed in Table 1.

Function	Tool Used	Vendor
VHDL Design, Compile, Test	ISE 5.2i	Xilinx
Software Design, Compile, Test	EDK 3.2	Xilinx
Logic and Software Simulation	MTI 5.6d	Mentor Graphics
Version Management	CVS	Open Source

Table 1 – Software used to develop, simulate, and manage the development environment of the Panoramic system

Test and Verification

When designing a project of this size, test and verification are of paramount importance to the successful completion of the project. The XDS team focused a large number of resources to ensure the design met Optos' requirements.

The team developed a full testbench that would fit around the design (shown in Figure 2). This testbench was designed to simulate every interface in the design. It consisted of a PowerPC microprocessor with peripheral cores to test protocols and communication interfaces. Both the design and the testbench used the swift model for the PowerPC microprocessor to run the software during simulation. A swift model is a cycle-accurate simulation model that can interpret PowerPC instructions and act accordingly.

Two software applications were used to test the system in simulation. The first ran on the testbench and the second ran on the system under test (SUT).

The function of the SUT application is to interact with the peripheral cores and to drive the inputs and outputs of the SUT. The application in the testbench interacts with the SUT in the following manner:

- Data received from the SUT is stored in memory.
- Protocol data is decoded, stored, and replied to, if appropriate.

The application for the SUT allows hardware design engineers to control many different features and to run a suite of tests in any order required. The XDS team also designed tests to verify the pure hardware interfaces without interaction with the applications.

All functionality was tested at unit, functional, and system level. The tests were all documented in the test and verification plan, which was approved by Optos before the system design phase began.

Conclusion

With Xilinx Design Services, you get a highly skilled and experienced team that will help get you to market faster by bridging the learning curve, delivering quality on-time designs, and working in close contact with your development team.

In a letter to the engineers at Xilinx Design Services, David Cairns, technical director at Optos, wrote, "I have been impressed with Xilinx Design Services. Their commitment to quality and high standards has eased transition into production. XDS went the extra mile, and we could not have achieved our design goals without their help."

For more information about Xilinx Design Services, visit *www.xilinx.com/xds*.

Creating One of the World's Highest-Resolution Digital Cameras

Phase One combined the low power consumption of the CoolRunner-II CPLD with the functional capabilities of the Virtex-II family.

The Phase One H25 digital camera back on a Hasselblad medium-format camera

©John Bendtsen

by Claus Mølgaard
Technical Manager – Hardware & Embedded Software
Phase One A/S
clm@phaseone.dk

From casual consumers to experienced professionals, digital photography has changed the way we all think about taking pictures. Indeed, digital cameras are replacing traditional film cameras at a faster rate than ever.

At the same time, digital image quality is improving quickly. Whereas consumer digital cameras typically have around 3 million pixels for an image, professional digital cameras have from 6 million to 22 million pixels.

As such, they present a far greater engineering challenge. Their large image file sizes must be moved quickly to achieve the high frame rates that professionals (such as fashion photographers) require. These cameras must also have advanced image processing capabilities and precise power management to extend battery life and reduce image distortion, which can be caused by thermal noise from high power dissipation of the built-in electronics.

At Phase One, we found a solution to these engineering challenges by incorporating the Xilinx CoolRunner-II™ CPLD and the Virtex™-II FPGA into the design of our digital camera back, the H25.

Modular Camera Design

Figure 1 shows the modular structure of a generic professional-quality, medium-format camera. This modularity allows photographers to not only swap lenses, but to use a different viewfinder or an alternative camera back.

A conventional camera back contains the type of film being used, but Phase One now produces a range of digital camera backs compatible with medium-format camera systems, enabling improved workflow and new creative opportunities. Compared to film, these camera backs also improve image quality and color reproduction.

Technical Requirements

At Phase One, we have created digital imaging solutions for professional photographers since 1993, supplying cameras for medical

diagnostics and aerial photography as well as studio and fashion photography. Through the years we have also supplied numerous OEM solutions for technically demanding applications. Our customers demand very high resolution, high sensitivity, ruggedness, reliability, a high continuous capture rate, and perfect color reproduction.

High-end professional film formats are larger than the 35-mm standard. For example, the classic 6 cm x 6 cm Hasselblad™ medium format is a favorite among professional photographers. Newer autofocus cameras from Hasselblad, Contax™, Mamiya™, and Fuji™ use a 6 cm x 4.5 cm format, which has also become popular. These larger film formats enable higher-resolution digital camera solutions without compromising image quality, which is highly dependent on pixel size. Larger pixels typically yield higher sensitivity and less noise.

With its 22 megapixels, H25 images enable an abundance of rich details.

Phase One digital camera backs are now available with image sensors ranging from 6 to 22 megapixels. This enables an active imaging area as large as 48 mm x 36 mm in our flagship H25 digital camera back.

The H25 image sensor is a huge piece of silicon that dwarfs even a large FPGA die. In fact, it is currently one of the world's largest commercially available imaging sensors, and helps the H25 push digital photography beyond the image quality of traditional film.

The specification for the H25 camera platform included the following key points:

- Huge image file sizes: 128 MB/image allows for 16 bits per color, for a total of 48 bits of red/green/blue (RGB) color depth
- Data moving capability as fast as 7 Gbps
- Big image buffer for long burst sequences
- Storage of uncompressed or losslessly compressed images in the RAW image format to keep the highest possible image quality
- High sustained frame rates with more than 30 frames/minute (faster frame rates are possible when images are captured in a burst)
- High processing power to enable severe image processing and compression
- 400 Mb IEEE1394 connection with proprietary high-speed image data streaming
- Advanced power management to reduce thermal dissipation as well as prolong battery life
- Small physical size.

Hardware Selection

When designing the H25, we used the CoolRunner-II CPLD to implement serial controllers, interrupt controller, and power management functions. We also used the plentiful on-chip resources of the CoolRunner-II device to implement the user interface controller, camera body interface, and glue logic.

The low current draw of the CoolRunner-II CPLD allowed us to minimize power dissipation, extending battery life and reducing the effects of thermal noise on image quality.

The powerful features embedded in the Virtex-II architecture provided even greater opportunities to maximize the performance of the H25, while also meeting important marketing targets.

Apart from restrictions on size, weight, and power, cost is always a concern. Time to market is paramount – the majority of our income from a new camera comes within the first year of its lifetime.

We considered alternatives, including ASIC and other "hard-coded" FPGA technologies. Although these were technically feasible, it was far more cost-effective, easier, and faster to complete the design using Virtex-II FPGAs.

Figure 2 outlines the functional blocks of the Phase One H25 digital camera platform.

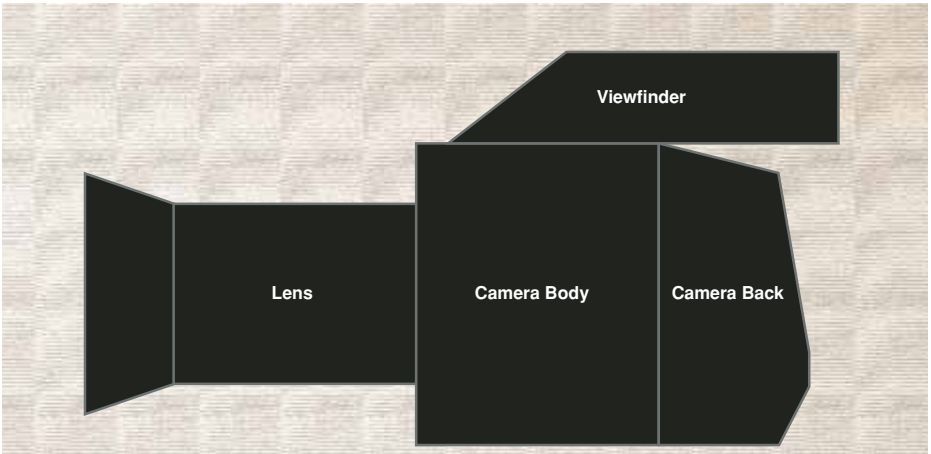


Figure 1 – Modules in a medium-format camera system

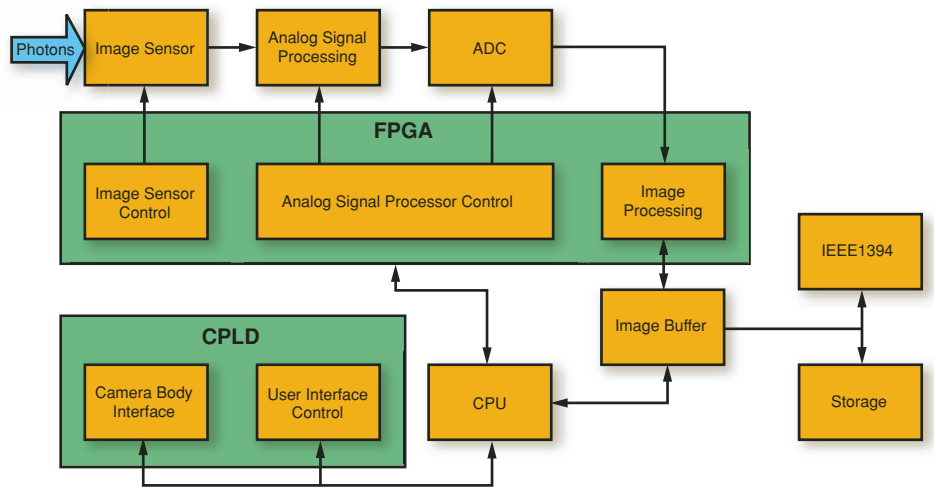


Figure 2 – Phase One digital camera platform

Implementing the H25 Camera Platform

We combined the high parallel image processing capability of the Virtex-II on-chip multipliers with the advanced digital clock managers (DCMs) and block RAMs to configure the Virtex-II FPGA as a powerful co-processsor chip. The Virtex-II device had ample capacity to offload most of the high data-rate tasks and image processing.

At the same time, we were able to implement large numbers of customized controllers within the device, thereby reducing the physical size. This also made it extremely easy to interface the central processor to various hardware blocks.

The DCMs are a huge benefit in systems where off-chip and on-chip frequencies are high. They enable easy and flexible de-skewing of board clocks and are very useful, for instance, in high-speed SDRAM controllers. No other FPGA vendor provides such an abundance of DCMs.

This is especially valuable in our application, because we use many different high-speed clocks when interfacing to different sections of the board. It also allowed us to apply a very fine-grained power management structure where various sections can be powered down or slowed to reduce the total power consumption. The H25 continuously adjusts frequencies on-the-fly to fulfill user demands at the lowest power consumption.

Virtex-II's SelectIO™+ technology supports high-speed signaling standards. We used these to move data onto and off the chip at 7 Gbps.

However, internal bandwidth is much higher if you consider all parallel processes that use local storage. We therefore used the Virtex-II block RAMs to implement various first in first outs (FIFOs), to handle data movement across different clock domains and on-chip local data caches. On-chip caches are absolutely essential for implementing high-speed image processing algorithms as well as advanced image compression.

By moving these functions into the Virtex-II FPGA, we also freed the main processor to execute real-time operating system (RTOS)-based system controller functions. This created a system performance overhead that can support additional functionality as market requirements increase. We can also upgrade products already in the field, adding new functions as we develop them.

Virtex-II's in-field reprogramming is crucial to this capability because it implements so much of the camera's functionality. This easy upgrade path also helps win loyalty from photographers, who see value in receiving ongoing improvements. We can keep ahead of our competition without having to release a totally new product every month. And we can assure our customers that they will always have a superior camera.

Conclusion

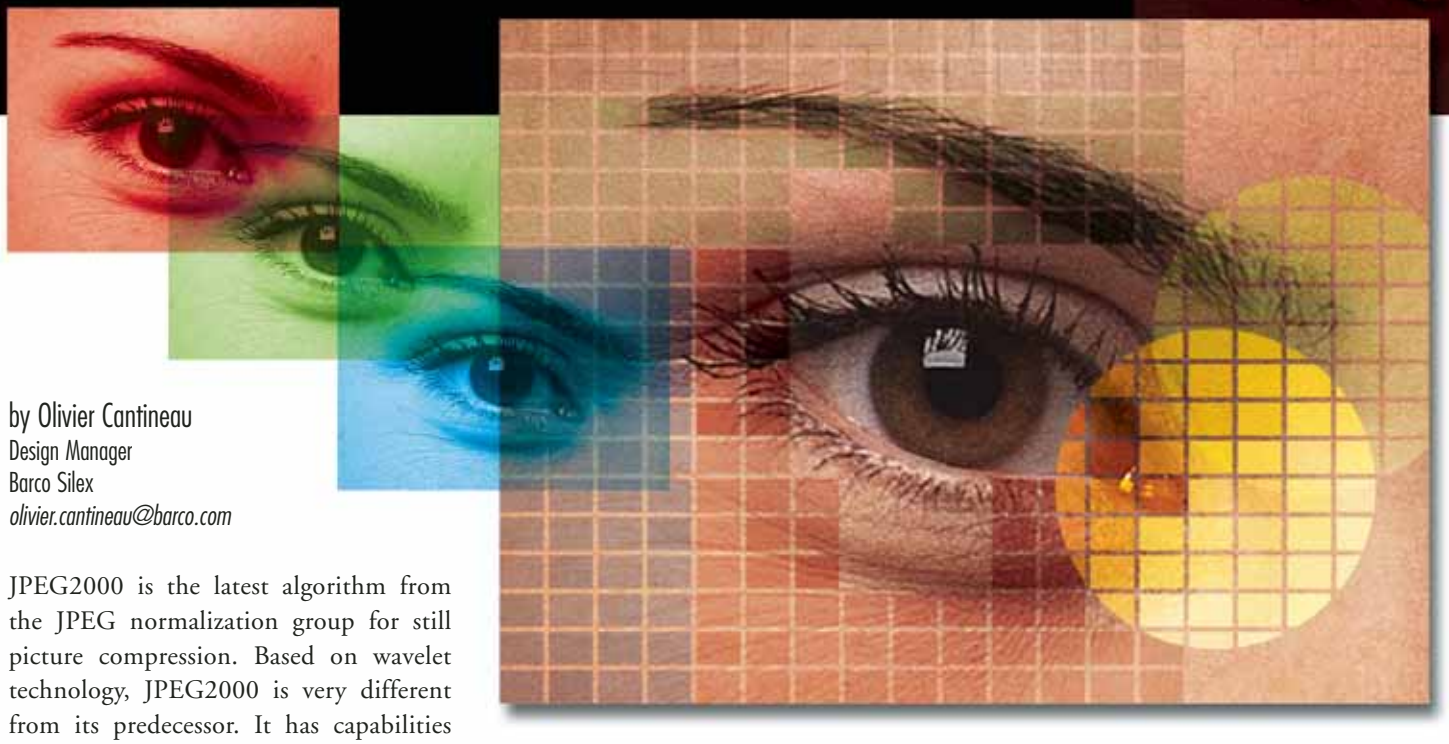
At Phase One we are committed to staying ahead of the competition, with excellent image fidelity in the field of digital high-end imaging. We hope to continue to push digital imaging beyond current technical boundaries.

The current trend towards breaking down the barriers between embedded software and hardware will help us to continuously exploit the technical capabilities of Xilinx FPGAs and CPLDs to achieve this ambitious goal. Visit our website at *www.phaseone.com* to see sample images and learn more about why photographers and other professionals choose our products.

A high-resolution model image captured with the H25

Accelerate JPEG2000 Compression

Barco Silex IP cores enable high-speed picture compression on Virtex-II and Virtex-II Pro FPGAs.



by Olivier Cantineau
Design Manager
Barco Silex
olivier.cantineau@barco.com

JPEG2000 is the latest algorithm from the JPEG normalization group for still picture compression. Based on wavelet technology, JPEG2000 is very different from its predecessor. It has capabilities that allow it to be adopted in a wide spectrum of applications, even extending to video encoding.

As a result, this compression scheme requires much more computational power than its classic JPEG predecessor. Furthermore, software implementations make poor candidates for applications requiring very fast encoding times.

Barco Silex has developed two JPEG2000 accelerator IP cores for high-performance applications: the BA112JPEG2000E encoder and the BA111JPEG2000D decoder. These cores handle the computationally intensive tasks of the JPEG2000 algorithm. Together with a host CPU, these cores create a complete JPEG2000 encoding or decoding solution.

Implementing the BA112JPEG2000E and BA111JPEG2000D cores on Xilinx Virtex™-II and Virtex II-Pro™ FPGAs paves the way to high-speed picture encoding applications with a very flexible architecture and shortened design time.

The JPEG2000 Algorithm

Although JPEG2000 is based on a single algorithm, it offers a wide range of tools for compressing and representing images. It is suitable for a large spectrum of applications ranging from Internet streaming to medical imaging to digital cameras.

JPEG2000 encompasses capabilities traditionally encountered in separate algorithms, such as:

- Lossy and lossless compression with excellent performance
- Precise compression ratio control with single-pass processing
- Bitstream progressivity, allowing consistent image previews with partial bitstream decoding
- Support for user-defined regions of interest in the image, encoded with higher quality than the other areas
- Error resilience.

Moreover, JPEG2000 compression quality outperforms the classical JPEG scheme for high compression ratios by generating fewer and less visible artifacts.

Let's explain the two consecutive processing stages required for JPEG2000 compression.

Tier 1 – DWT-Based Compression

The JPEG2000 algorithm divides the image into rectangular tiles of a configurable size.

Each tile undergoes a two-dimensional discrete wavelet transform (DWT), which reorders the tile's frequency information into a series of pictures (subbands). A subband results from the filtering of the original tile for a given frequency range.

A parameter of the transform (the number of decomposition levels) defines the number of frequency intervals.

Each subband can then undergo selective quantization by a programmable factor

for lossy compression. Bypassing the quantization yields lossless operation.

The algorithm further divides the resulting quantized subbands into smaller rectangular blocks (code blocks).

A modeler examines the bit planes of the current code block, beginning with the most significant one. In each plane, it scans the bits in a zigzag order and determines their context by using information such as the predominant value of the surrounding bits.

Finally, an arithmetic encoder processes the value of the bit and the context. It generates a code stream representing the compressed code block.

This arithmetic encoder also computes distortion metrics, which reflect the image distortion that would be encountered when reconstructing the code block with its currently encoded portion.

Tier 2 – Packet Selection and Reordering

The code stream generated by the arithmetic encoder, together with the distortion metrics, allows JPEG2000 to selectively build the final bitstream at the post-processing stage. This process is driven by two user-defined parameters:

- Compression ratio – This Tier-2 stage selects incoming packets to attain a user-specified compression ratio. The algorithm rejects packets that do not sufficiently improve the compression distortion. This mechanism allows precise control of the generated compressed file size, while maintaining good image quality.
- Progression order – JPEG2000 allows an initial preview of a picture with the first portion of the bitstream. Decoding subsequent parts of the compressed file progressively refines the image. JPEG2000 also standardizes various refinement orders by prioritizing an image characteristic, such as quality or resolution. The Tier-2 stage achieves the desired progression order by reordering incoming packets.

JPEG2000 Implementation

Because of its powerful features, JPEG2000 requires more computational resources than the classic JPEG to achieve similar encoding and decoding speeds.

The features of Xilinx Virtex-II series FPGAs make them an excellent choice for implementing JPEG2000 solutions, including fast and numerous RAM blocks and a large amount of logic resources.

RAM blocks allow the implementation of a large on-chip tile buffer, increasing the core's overall performance and integration level.

Moreover, thanks to their on-chip IBM™ PowerPC™ 405 cores, Virtex-II Pro devices allow you to construct a complete JPEG2000 system. Indeed, the IP cores, implemented in the programmable logic, will execute the first stage of the JPEG2000 algorithm (wavelet transform, quantization, and entropy encoding). A software routine running on the host processor will more suitably execute the Tier-2 stage.

Figure 1 shows a block diagram of the Barco Silex BA112JPEG2000E IP core. This illustrates the main functional modules and a simplified view of the interfaces.

You can input pixel data through the pixel interface and get data streams at the compressed interfaces together with distortion metrics. The core features a simple, generic CPU interface, suitable for interfacing as a bus peripheral to various processors, including a PowerPC system.

As an example, Virtex-II devices can be used together with the BA112JPEG2000E and BA111JPEG2000D cores to perform picture encoding or decoding with timings compatible with NTSC (National Television Standards Committee) requirements. In other words, images have a resolution of 720 pixels by 480 lines in 4:2:2 color format within 33 ms. The cores require 8,200 slices and 54 RAM blocks for an irreversible solution (lossy), or 11,500 slices and 66 RAM blocks for reversible compression (lossless).

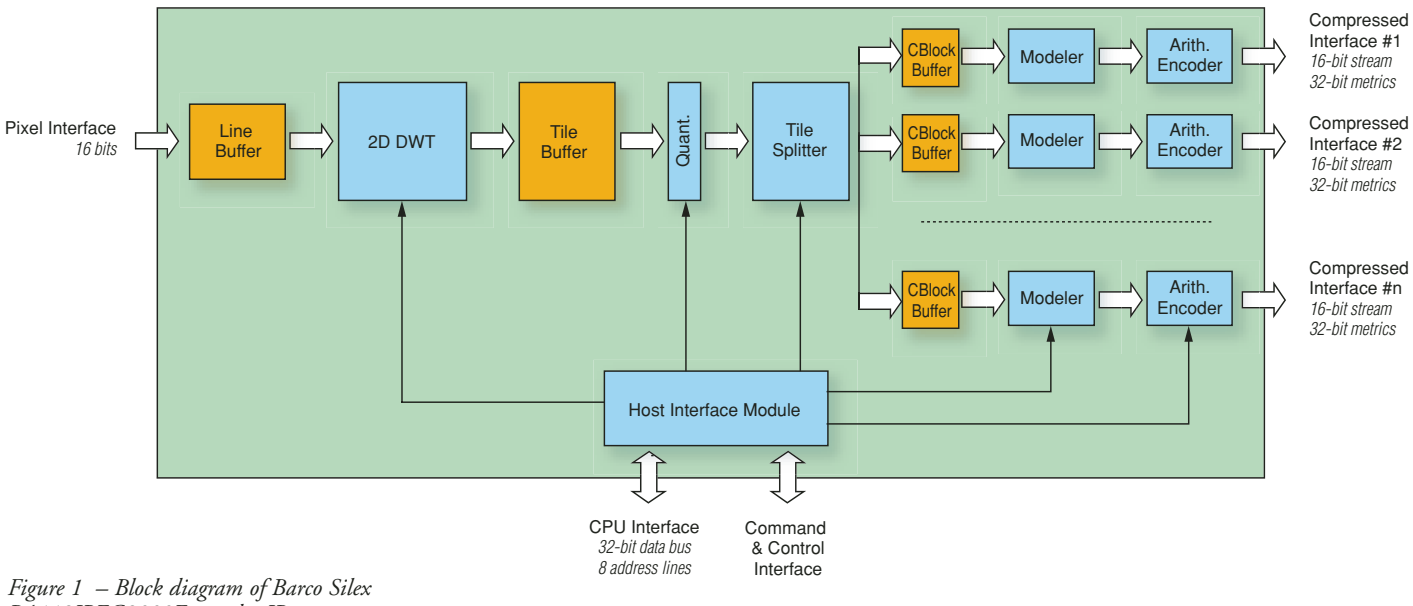


Figure 1 – Block diagram of Barco Silex BA112JPEG2000E encoder IP core

2-D DWT

The first module of the BA112JPEG2000E core is the DWT engine. This module can be configured to accept tiles of pixels as large as 128 by 128. It performs two-dimensional discrete wavelet decomposition on incoming data with as many as five programmable decomposition levels. The wavelet transform can be programmed as lossy, lossless, or bypassed.

The DWT module accepts incoming pixels of any size – up to 10 bits for lossy and up to 12 bits for lossless. It stores its results in the on-chip tile buffer to undergo quantization and code block decomposition.

Quantizer

The quantizer fetches the subbands available from the tile buffer and applies a programmable quantization step. Different quantization steps are programmable for each subband. Thus, you can weight lower frequency subbands differently than higher frequency ones.

You can also bypass the quantizer for lossless mode.

Tile Splitter

This unit further divides quantized subbands into rectangular code blocks of a programmable size (as large as 32 by 32 pixels) in preparation for entropy encoding by an arithmetic encoder.

The BA112JPEG2000E and BA111JPEG2000D cores feature a configurable number of entropy encoders, placed in parallel, to sustain high encoding rates. You can select the number of implemented chains during the IP synthesis process. Each entropy chain processes a code block independently of neighboring chains.

The tile splitter module arbitrates between the available chains, dispatching the various code blocks to be encoded. It stores code blocks in local code block buffers.

Modeler and Arithmetic Encoder

The modeler performs the first part of entropy encoding. It examines the code block bit plane by bit plane and extracts relevant bits in zigzag order for each bit plane. The modeler then computes the context information needed by the arithmetic encoder.

The arithmetic encoder processes the bits and contexts, and makes the stream and the distortion metrics available at the compressed interface. These are used by the Tier-2 part of the JPEG2000 algorithm.

Host Interface Module

This module allows the interfacing of the core to a CPU: It contains configuration registers for the various modules and gives status information about the encoding progress.

The host interface module also features a separate command-and-control interface that allows fast control of the IP core with minimal or no CPU intervention. Hence, the command-and-control interface makes it possible to build a system in which a small amount of logic drives the BA112JPEG2000E and BA111JPEG2000D cores, which are not directly connected to a host CPU. This increases the integration flexibility of the IP cores.

Conclusion

Barco Silex's BA112JPEG2000E and BA111JPEG2000D IP cores are targeted to high-speed JPEG2000 encoding and decoding, providing access to the wide range of capabilities now possible with the JPEG2000 standard.

This new standard defines an algorithm able to offer a large spectrum of features, such as progressive bitstream, precise rate control, region of interest, and high-quality joined lossy and lossless compressions.

This rich set of advantages turns JPEG2000 into an important actor in the compression world, especially as it is not limited to still picture compression.

The computational complexity of the JPEG2000 standard requires hardware platforms for high-speed applications compatible with real-time video encoding. Barco Silex BA112JPEG2000E and BA111JPEG2000D IP cores, acting as compression accelerators on Xilinx Virtex-II and Virtex-II Pro Platform FPGAs, can give you the speed you need.

For more information about Barco Silex IP cores, visit *www.barco-silex.com.*

Get Award-Winning Support at Your Fingertips

Find the answer you need, when you need it, with MySupport.

THE ASSOCIATION OF SUPPORT PROFESSIONALS
Award Winner 2003
The Year's Ten Best Web Support Sites

by Doug Horne
Product Marketing and Web Development, Global Services Division
Xilinx, Inc.
doug.horne@xilinx.com

A robust online resource, MySupport stands ready to answer all of your design questions 24 hours a day, seven days a week, 365 days a year. Our award-winning support website allows you to personalize content so that you see only the information that's relevant to you.

As shown in Figure 1, MySupport offers access to sections such as Answers Search, Documentation, Tech Tips, Forums, Problem Solvers, TechXclusives, WebCase, Software Updates, and MyAlerts; you can get the answer you need when you need it.

Figure 1 – A sample MySupport landing page displays the Support News and Xilinx Global Services windows.

What Are You Looking For?

- Answers Database: With new and improved search technology, answers database searches are faster and more accurate than ever. Our advanced search or answer browser tools enable you to easily access the latest answers

to technical issues from a huge database of more than 4,000 answers, indexed by logical category.

- Documentation: Your one-stop source for software manuals in PDF, compressed PDF, or search-enabled HTML format. This section also includes application notes, data sheets, and more. You can even browse publications by part.
- Tech Tips: Get the latest technical information about development tools, device families, interface tools, and hardware and IP solutions.
- Forums: Collaborate with other designers in discussion groups or chat rooms; join the popular newsgroup *comp.arch.fpga*.
- Problem Solvers: Get instant help with installation and configuration, PCI applications, and JTAG implementation. This interactive tool uses a series of questions to diagnose and troubleshoot your configuration or installation problem automatically, saving you hours of work.
- TechXclusives: Expert Xilinx application engineers keep you informed of the hottest issues and latest techniques.
- WebCase: Use our WebCase to manage hotline cases. You can check status, add notes, and even close a case on the Web, anytime.

Let's Get Personal

MySupport puts powerful support tools to work for you. As shown in Figure 2, these features include:

- Support News: Get the latest in support-related news and developments on Xilinx products. TechXclusives are also featured in this section.
- Xilinx Global Services: Here you'll find all the latest news on the Xilinx Global Services Division's offerings.
- MyAlerts: Stay current with updates to technical publications, software updates, products, Tech Tips, and design and system resources with MyAlerts. Whenever you log in, you will automatically be notified of new downloads (such as service packs and IP update releases) relevant to your design.

Figure 2 – Various windows within MySupport

- MyXPA: With MySupport's newest feature, Xilinx Productivity Advantage (XPA) customers can check their training credits, sign up for classes, and more.
- What's New: View the latest products, devices, design tools, and system solutions from the Xilinx product catalog.
- Latest Answers: View the answer records you've visited, make note of the most recent answers published by Xilinx engineers, and see what others are saying.
- Product Genealogy: Get quick access to fabrication, assembly, and die-related parts change notice (PCN) information.
- My Bookmarks: Create links to your favorite websites and answer records, or search queries on MySupport.
- Stock Quotes: Watch your favorite stock ticker symbols, including latest price, change, and last trade information.

Personalization even extends to your preferred language (Figure 3).

MySupport makes it easy to stay on top.

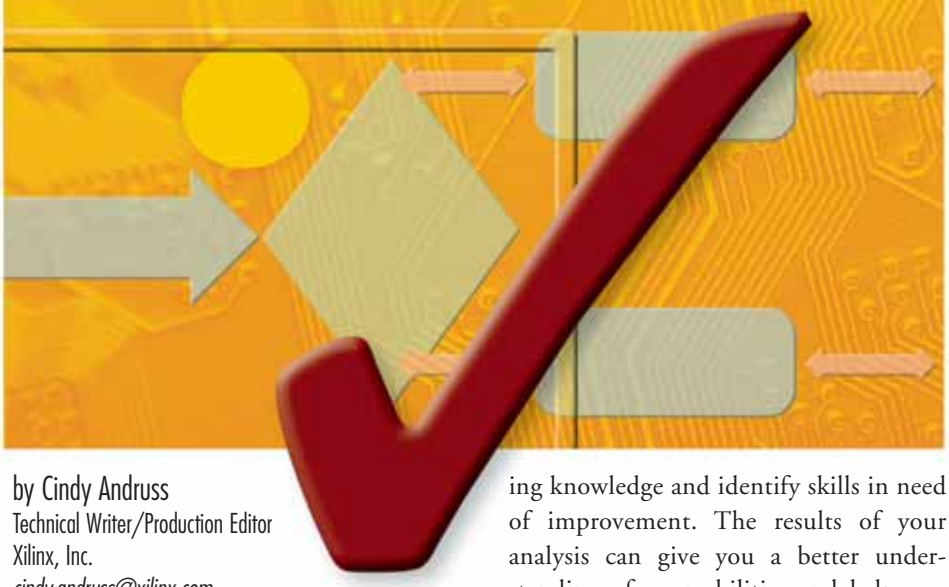
Figure 3 – Located in the bottom left-hand corner of the main landing page, Japanese and Chinese language translation is provided within MySupport.

Do It Today

"Finish Faster" by taking advantage of these powerful tools. Registration is easy and free. To sign up for MySupport, visit *www.xilinx.com/support* or click on the "Support" button on the top navigation bar anywhere on *www.xilinx.com*. If you have any questions, contact the author at 408-626-6317 or e-mail *doug.horne@xilinx.com*.

Customize Your Education with Skills Assessments

By determining which Xilinx course offering is most appropriate, you can maximize your learning time and minimize training costs.



by Cindy Andruss
Technical Writer/Production Editor
Xilinx, Inc.
cindy.andruss@xilinx.com

Xilinx Education Services offers a variety of instructor-led courses, e-Learning modules, and curriculum paths for your design specialty to help you increase your skill set and create efficient designs. The diversity of courses and delivery methods empowers you with the flexibility to choose the learning method that best meets your needs and preferred learning style.

But with so many training options, how do you determine which courses will enhance your present design capabilities? Take a Xilinx skills assessment.

Available at no charge, Xilinx skills assessments measure your current training knowledge and identify skills in need of improvement. The results of your analysis can give you a better understanding of your abilities and help you make wiser registration decisions as to which courses or modules best fulfill the gaps identified.

"Efficiency is the name of the game these days as companies strive to lower costs and maximize their investments," said Patrice Anderson, course development manager at Xilinx Education Services. "When you take a Xilinx skills assessment, you lower your training costs and maximize the return on your training investment by taking only the courses you need. By completing a skills assessment, you are able to develop a personalized learning plan that will get you on the fast track to implementing Xilinx solutions."

Knowledge Drives Effective Learning

With your personalized learning plan, you can enroll only in the Xilinx courses that are right for you. Xilinx courses will show you how to reduce time to knowledge, boost system clock speeds, fit more functionality into smaller devices, shorten design cycles, maximize productivity, and increase your skill set.

Jahan Lotfi, a design engineer at Cortina Systems™, uses skills assessments to prepare for class. "I always believed that once you 'know the things you don't know,' and then sit in the class, you pay more attention than when you are just presented new material," Lotfi said. "I think sometimes you just don't know the significance of the material presented to you until you have thought about it a bit and analyzed its advantages.

"As a result of taking the skills assessment, I was a better student and came out of the class more informed."

Plan for Lowest Cost Training

Although engineers can use skills assessments to organize a learning strategy for themselves, their managers are also finding skills assessments a helpful planning tool.

When manager Don Meyerhoff of Standard Microsystems™ needed a resource to help plan the training budget for his staff, he turned to Xilinx Education Services Representative Mike Weir for assistance. Weir pointed Meyerhoff to skills assessments.

"We used to team up the customer with a Xilinx instructor on the phone to determine the customer's skill level, or we e-mailed the course syllabus or pointed the customer to the online course description," Weir said. "But now we can offer a much more effective solution."

By asking his engineers to take online skills assessments, Meyerhoff can gauge which skill levels they're at and ultimately, which courses are or aren't necessary. As a result, his training budget is more precise.

Beyond helping you choose appropriate single courses or online modules, managers can use skills assessments to help them budget the appropriate amount of training credits included in Xilinx Productivity Advantage (XPA) agreements. The XPA Solution includes education, support services, soft-

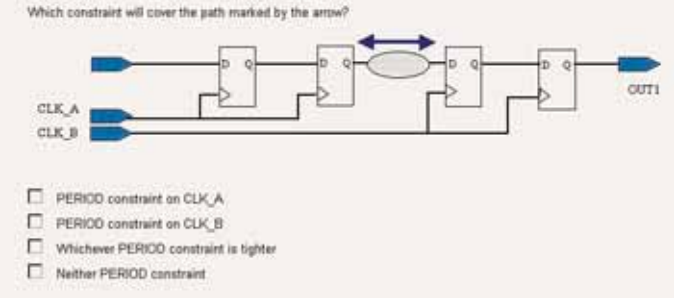


Figure 1 – A multiple-choice question from the Designing for Performance skills assessment

ware, and IP cores in one comprehensive package. Training credits are included in every XPA and can be redeemed toward any Xilinx Education Services offering.

To help determine an adequate number of training credits, managers can ask their engineers to complete skills assessments to determine which courses they should take going forward, and thus how many training credits to include.

Completing an assessment also provides the basis for a qualified training plan – one that can be tailored to individual companies based on their needs.

The Skills Assessment Advantage

To determine your current Xilinx design skill level, simply go to *https://xilinx.onsaba.net/xilinx*. Select the skills assessments respective to the courses for which you want to assess your competence.

Of course, there is no obligation to register for any course, and you can skip the assessment and register for a course without it. But you can also test your skills as often as you like to help better determine whether you would benefit from attending a course.

Xilinx also recommends that you reassess your skills after completing a course, comparing your post-course score with your pre-course score. Lotfi believes that taking the skills assessment again some time after completing an FPGA course helped him discover new tool design capabilities.

"The [skills assessment] showed me that there were some tool capabilities I didn't know about because I had not come to need them in previous designs," he said. "I found these [tool capabilities] from the [skills assessment] questions."

Most of the skills assessments include matching, multiple choice, and fill-in-the-blank exercises and take about five to 15 minutes to complete (for a sample question, see Figure 1). A score of 80 percent or higher implies you are familiar with at least 80 percent of the course content and would not benefit significantly from taking the course. On the other hand, if you are unable to answer at least four out of every five questions correctly, you should attend the class to increase your individual productivity. Ultimately, the decision is yours.

Your score is also yours. Only you have access to your skills assessment results. To view your score, simply use your username and password when logging in to the Learning Catalog.

Conclusion

Efficiency goes beyond design. It begins with effective training and plays an important role in meeting budget constraints and tight delivery schedules. With up-to-date FPGA design skills, you are in a better position to maximize your investment in Xilinx solutions and reduce your time to market.

Now you can make the most of your learning time and training investment with Xilinx skills assessments. Enroll only in the training you need to get the skills and knowledge that will bring your design to market fast.

For more information about Xilinx skills assessments, visit *https://xilinx.onsaba.net/xilinx*, or contact the Xilinx Education Services registrar via e-mail at *registrar@xilinx.com* or by telephone at (877) XLX-CLAS (2627). For a list of frequently asked questions, go to *www.xilinx.com/support/training/skills-assessment.htm.*

To learn more about the Xilinx Productivity Advantage Program, visit *www.xilinx.com/xpa*, or contact an XPA representative at (800) 888-FPGA (3742) or e-mail *fpga@xilinx.com.*

Xilinx Wins Architecture Awards for Colorado Office Complex

The architects' emphasis on native architecture and sustainable materials has received national recognition.

by Forrest James
Director of Corporate Facilities
Xilinx, Inc.
forrest.james@xilinx.com

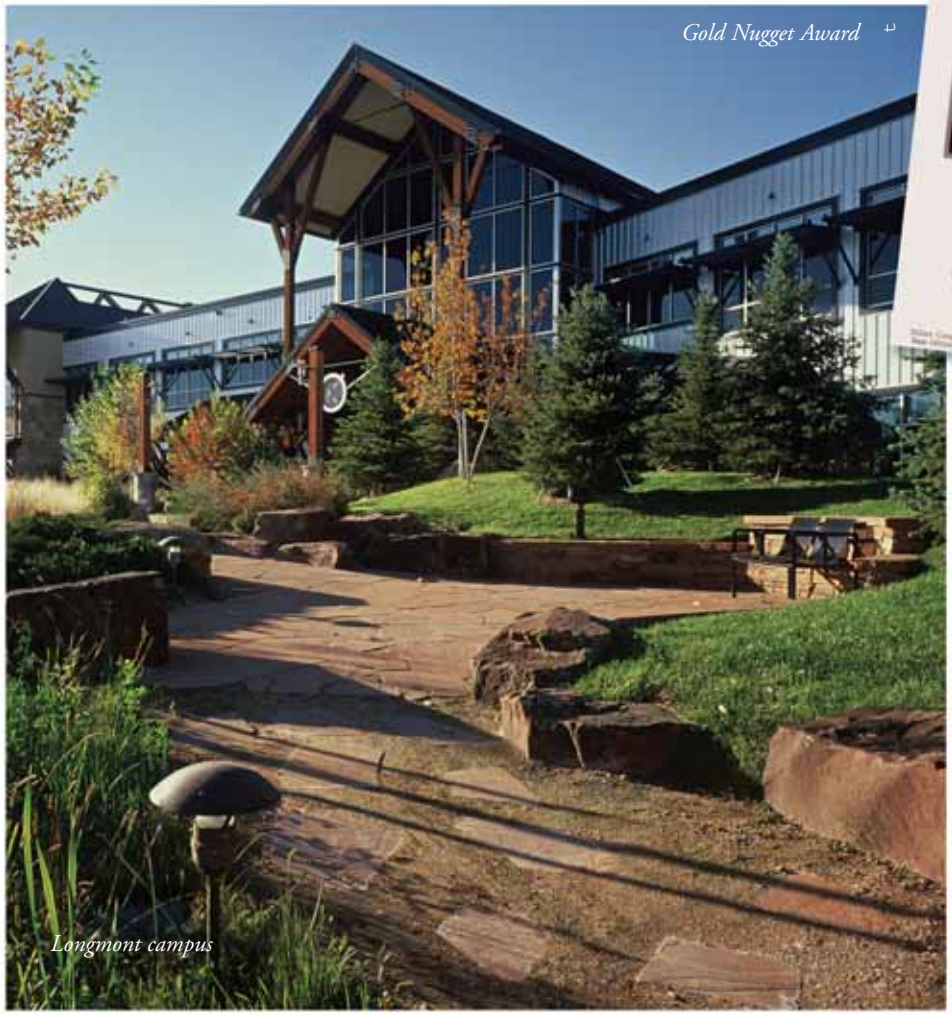
When *FORTUNE* magazine ranked Xilinx as fourth on its 2003 list of the 100 Best Companies to Work For™ in America, the editors based their selection on a range of criteria, including benefit packages, number of minority staff, and layoffs – or lack thereof.

But Xilinx employees at the company's Longmont, Colorado, campus can add to that list a literal "best place to work in." Their new office building, opened in April 2002, has garnered several awards for its energy conservation and building design.

In June 2003, DTJ Design (the landscape architect, design architect, and master planner) and the Neenan Company (the architect of record and general contractor) received a Gold Nugget Award for the Best Office/Professional Building (<60,000 square feet) at the annual Pacific Coast Builders' Conference.

The Longmont office has also received acclaim from the Colorado Renewable Energy Society, the Colorado Chapter of the American Institute of Architects, and the Colorado Society of Landscape Architects.

Gold Nugget Award


Longmont campus

Associate Len Segel of DTJ Design says, "We didn't start out with the idea to create an award-winning building. Yet it was a really special project [for us] because it was a close collaboration with an enlightened client interested in innovation, flexible work stations, and in particular paying attention to employees' needs."

Indigenous Innovations

The Colorado landscape and Rocky Mountains backdrop served as the inspiration for many design elements. The architects fit one of three interconnecting buildings, or "segments," into the side of a hill. The segments themselves are named "Prairie," "Foothills," and "Mountains" – all following the natural curvature of the topography.

The structure combines diverse architectural styles from plains, agricultural, and mountain-style dwellings, with arching canopies of gabled roofs and trusses, local

quarry stone, rough-hewn lumber, and wooden braces. Snow slides right off the metal roof. A half-silo in the cafeteria mimics nearby agricultural structures. And a separate, 6,000-square-foot retreat center, accessible by a bridge from the upper level, has a lodge-like feel.

A winter garden in the central lobby features plants in a temperature-and-humidity controlled environment. Outdoors, the landscaping is indigenous to the native Colorado environment, including drought-tolerant plants and native rock.

The 100-acre building site also includes a "wet meadow," which remains as a natural feature. It continues to attract wildlife such as rabbits, birds, prairie dogs, and – to the delight of the employees – a family of foxes.

A Green Bill of Materials

Inside, sustainable and recycled materials literally run from floor to ceiling. The stairway in the entrance lobby is made of bamboo wood (actually a grass), and the winter garden features laminated timber trusses from young rather than old-growth forests.

Floors are linoleum, a natural material, or stained concrete, which is energy-efficient. Some of the interior walls comprise pressed, recycled sunflower seeds, which make for a visually appealing pattern. Cubicle walls, conference tables, even chair fabrics are all partially or entirely recycled material.

The lighting is particularly innovative – and cost-effective. A miniature lighting optical shelf (MOLS) system casts exterior daylight into the building's interior; high-intensity, low-consumptive lighting then dims automatically to maintain a comfortable lumen level at the desktop.

To encourage employees to leave their cars at home and bike to work, the campus includes both indoor bike storage and an external rinsing station for muddy days. Biking paths throughout the perimeter of the campus connect to Boulder County biking paths.

Working Out the Details

The Longmont office is largely dedicated to programmable logic software development. Emulating that concept of reconfigurability, the building's designers created flexible working spaces so that departments can be organized and reorganized to adapt to future business needs.

The bottom level includes lab spaces and infrastructure and support facilities, such as server rooms. A gym and cafeteria are also located on the bottom floor.

Both the lower and upper levels are built upon a system of raised flooring (and modular carpeting) so all of the air conditioning vents, network cables, and electrical wiring are contained beneath the floor rather than above. Thus, cubicle spaces can be configured into any shape, unencumbered by ducts or power poles.

The flooring system allows employees to control the airflow within their own cubicles, which according to Segel also has a cost benefit: "Rather than having to travel all the way from the ceiling – and fight any rising hot air as well – cool air or heat is delivered at the floor, closer to where the employees are sitting."

Conference rooms, break areas, and restrooms are located in the center of each of the three building segments. The break rooms are furnished with high tables, bar seats, and a whiteboard for sudden brainstorming: "Innocent conversations can turn into opportunities for innovation," says Segel.

Window offices are nonexistent at Xilinx. Everyone has access to the striking views of the surrounding landscape. Some rooms boast even more spectacular visuals and amenities, such as fireplaces and CD players. They are designed for informal meetings, relaxation, or both.

Room to Grow

Looking to the future, Xilinx and its architects drew up a master plan for building an additional one million square feet of office space – 10 buildings to accommodate as many as 4,000 more employees.

Reflecting on the completed portions of the project and those still yet to come, Segel explains that the overall theme will remain faithful to Colorado. "It feels like it's a natural Colorado-type building, and that was very important to employees and to management – that it wouldn't feel like a California facility but a Colorado facility."

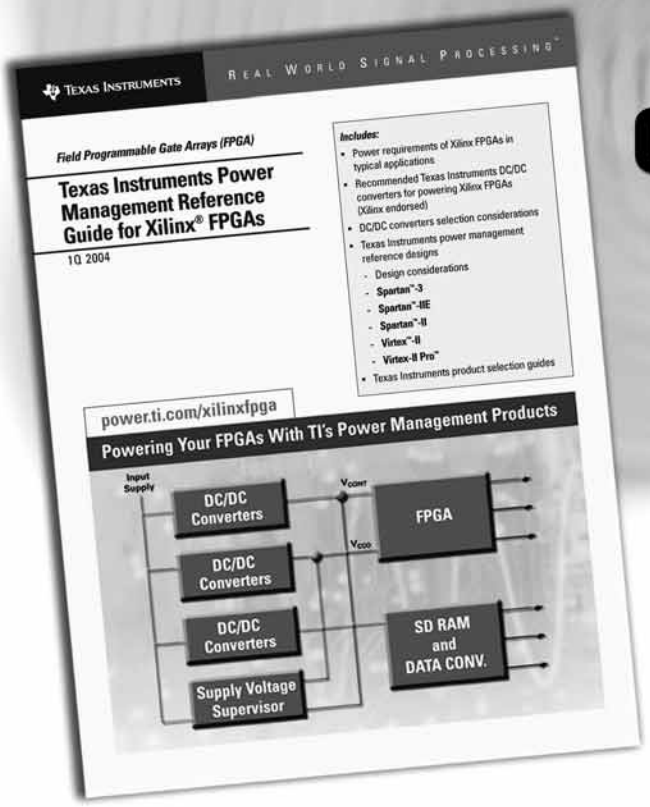
"We also tried to have the building be a recruiting tool to bring new employees to Xilinx and to retain employees with a great working environment," Segel concludes.

Photos courtesy ©Ed LaCasse Photography

LEAD BY Memec DESIGN

Need power solutions for your next design?

Together we wrote the book on powering FPGAs.



Powering Xilinx FPGAs just got easier with Texas Instruments' power management products. A free copy of the TI reference guide, developed with the help of our top FPGA and power management experts, has been included in this issue. Success is at your fingertips. Let **Insight Memec** take you there.

Call today to receive TI's TPS40K™/SWIFT™ CD for FREE!

Insight MEMEC

Texas Instruments AUTHORIZED DISTRIBUTOR

XILINX® The Programmable Logic Company℠

Contact Insight Memec today at **800.677.7716** or visit our website at ***www.insight.na.memec.com***.

Field Programmable Gate Arrays (FPGA)

Texas Instruments Power Management Reference Guide for Xilinx® FPGAs

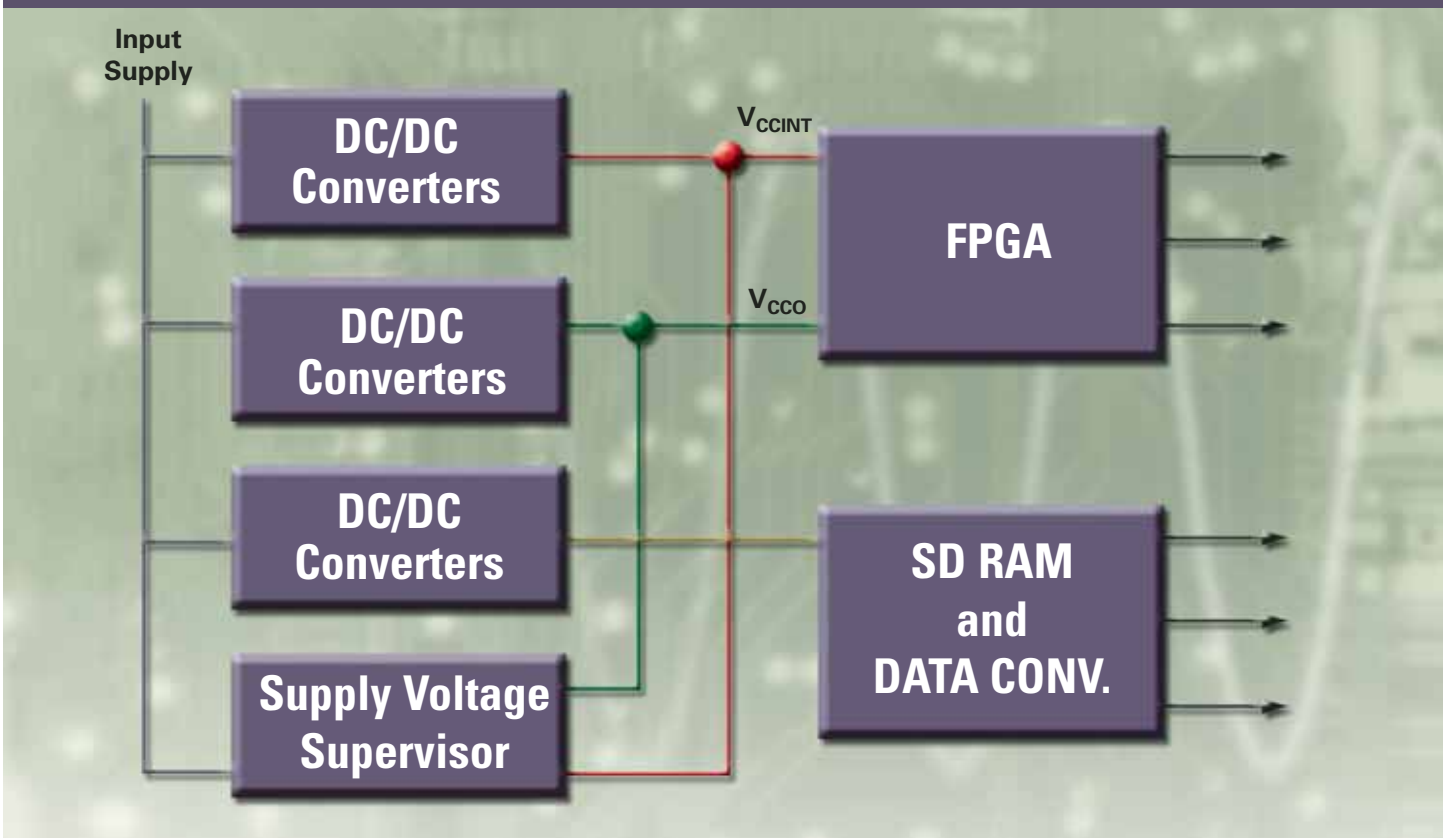
1Q 2004

Includes:

- Power requirements of Xilinx FPGAs in typical applications
- Recommended Texas Instruments DC/DC converters for powering Xilinx FPGAs (Xilinx endorsed)
- DC/DC converters selection considerations
- Texas Instruments power management reference designs
  - Design considerations
  - **Spartan™-3**
  - **Spartan™-IIE**
  - **Spartan™-II**
  - **Virtex™-II**
  - **Virtex-II Pro™**
- Texas Instruments product selection guides

power.ti.com/xilinxfpga

Powering Your FPGAs With TI's Power Management Products



Power Requirements of Xilinx® Solutions in Typical Applications

This information is intended to provide the designer with a general understanding of the power requirements of Xilinx FPGA families in typical applications. Please refer to the Xilinx Power Estimators, available at **www.xilinx.com/power**, for closer approximations specific to individual FPGA devices and applications.

Power Requirements of Xilinx FPGA Families

	Spartan™-II	Spartan™-IIE	Spartan™-3*	Virtex™-II	Virtex-II Pro™
V_{CCINT} (core)	2.5 V +5% @ 200 mA to 2 A	1.8 V +5% @ 200 mA to 2 A	1.2 V +5% @ 200 mA to 5 A	1.5 V +5% @ 300 mA to 10 A	1.5 V +5% @ 300 mA to 10 A
V_{CCO} (I/O)	3.3 V, 2.5 V and/or 1.5 V +5% @ 50 mA to 500 mA	3.3 V, 2.5 V, 1.8 V and/or 1.5 V +5% @ 50 mA to 500 mA	3.3 V, 3.0 V, 2.5 V, 1.8 V 1.5 V and/or 1.2 V +5% @ 50 mA to 1.5 A	3.3 V, 2.5 V, 1.8 V and/or 1.5 V +5% @ 50 mA to 3 A	3.3 V, 2.5 V, 1.8 V and/or 1.5 V +5% @ 50 mA to 3 A
V_{CCAUX}	—	—	2.5 V +5% @ 300 mA (max)	3.3 V +5% @ 300 mA (max)	2.5 V +5% @ 300 mA (max)
Digital Voltages above, Analog voltages for RocketIO Multi-Gigabit Transceiver (MGT) below.					
$AV_{CCAUXTX}$	—	—	—	—	2.5 V +5% @ 60 mA/MGT
$AV_{CCAUXRX}$	—	—	—	—	2.5 V +5% @ 35 mA/MGT
AV_{TTX}	—	—	—	—	1.8 V to 2.625 V @ 15 mA/MGT
AV_{TRX}	—	—	—	—	1.8 V to 2.625 V @ 30 mA/MGT

**Spartan-3 information is based on the advance datasheet and is current as of December 2003.*

TI's Power Reference Designs for Xilinx FPGAs

Complete schematics, bills of materials and additional designs available at:

power.ti.com/xilinxfpga

Questions? fpgasupport@list.ti.com

Important Design Considerations:

- The shown power solutions are for applications guidance. Please visit the TI website for the latest updated information.
- Although Xilinx FPGAs do NOT require it, TI's reference designs employ sequencing, as well as a Supply Voltage Supervisor (SVS) to monitor the input supply. This practice is consistent with good power supply design and prevents the input power supply from being pulled down due to in-rush currents for charging large capacitive loads.
- V_{CCAUX} powers time-critical resources in the FPGA, including the Digital Clock Managers (DCMs). Therefore, this supply voltage is especially susceptible to power supply noise. V_{CCAUX} can share a power plane with V_{CCO}, but only if V_{CCO} does not have excessive noise. Changes in V_{CCAUX} voltage beyond 200 mV peak-to-peak should take place at a rate no faster than 10 mV per milli-second.
- These designs meet Xilinx's V_{CCINT} and V_{CCO} startup profile requirements, where applicable, including monotonic voltage ramp, in-rush current and power voltage ramp time requirements.
- The designs show the most common V_{CCO} voltages, 2.5 V or 3.3 V. Other voltages can be supported by interchanging the depicted DC/DC converter with devices from the same family, assuming that the linear regulator does not exceed its power dissipation rating. Refer to the selection guides on pages 100 - 11 for alternatives.
- Only the minimum input and output capacitors for each IC are given in the schematics. Larger bulk and/or bypass capacitors will be required between the input supply and DC/DC converters depending on the placement of the input supply relative to the converters. Each FPGA also requires a minimum amount of bypass capacitance on each power rail as specified by Xilinx.

TI Recommended DC/DC Converters for Powering Xilinx® FPGAs

	Digital Voltages				Analog Voltages
	Low Dropout (LDO) Linear Regulators to 3 A	Switching DC/DC Converters (Integrated FET) to 9 A	Switching DC/DC Controllers (External FET) to 20 A	Switching DC/DC Modules to 30 A	Low Dropout (LDO) Linear Regulators to 1.5 A
Spartan™-II	TPS79xxx TPS786xx TPS703xx UC382-x	TPS54xxx	TPS6420x TPS400xx	PTH Series	—
Spartan™-IIE	TPS79xxx TPS786xx TPS703xx UC382-x	TPS54xxx	TPS6420x TPS400xx	PTH Series	—
Spartan™-3	TPS79xxx TPS786xx TPS704xx UC382-x	TPS54xxx	TPS6420x TPS400xx	PTH Series	—
Virtex™-II	TPS79xxx TPS786xx	TPS54xxx	TPS6420x TPS400xx	PTH Series	—
Virtex-II Pro™	TPS79xxx TPS786xx	TPS54xxx	TPS6420x TPS400xx	PTH Series	TPS79xxx TPS786xx

TI Linear Regulators require additional circuitry in order to provide soft-start on power-up.
These recommended DC/DC converter products have been tested and endorsed by Xilinx to power the listed FPGAs.
Digital voltages may be powered by either linear or switching DC/DC regulators. Analog voltages require linear regulators for minimal noise.
Please see the selection guides on pages 10 and 11 for electrical specifics of the recommended devices.

DC/DC Converter Selection Considerations

- **Low Dropout (LDO) Linear Regulators**
  - Easiest, smallest, and most cost-effective type of DC/DC converter to implement.
  - Typically just requires the addition of a small input capacitor and output capacitor for stability.
  - Exhibits low noise and therefore are ideal for powering analog voltages.
  - Recommended for low power applications only due to generally low efficiency (LDO Eff = V_{OUT}/V_{IN} x 100%) and resulting heat. Ensure the application does not violate the regulator's maximum allowable power dissipation. Refer to TI Application Note SLVA118 ***'Digital Designer's Guide to Linear Voltage Regulators and Thermal Management'*** for guidance.
  - Low dropout performance needed to support small V_{IN} to V_{OUT} voltage differential.

- **Switching DC/DC Converters**
  - Easiest, smallest, discrete IC, switching DC/DC solution to implement due to integration of the FETs.
  - Requires the addition of an inductor and capacitors for the output filter.
  - Use fixed output voltage, internally compensated devices to minimize component count and simplify implementation.
  - Using adjustable output voltage TPS54xxx devices instead of fixed output options allows for external compensation, resulting in a smaller solution due to the ability to minimize the inductor and use ceramic capacitors.
  - Exhibits switching noise.
  - Up to 95% efficient, giving a much cooler solution than a linear regulator, but due to the power dissipation in the internal FETs, switching DC/DC converters support only up to about 9 A.

- **Switching DC/DC Controllers**
  - More cost-effective than DC/DC converters, but consume more board space and are more challenging to implement as they require the addition of external FETs.
  - Can support high-current applications, limited only by the controller's drive and the external FET's power dissipation capabilities.

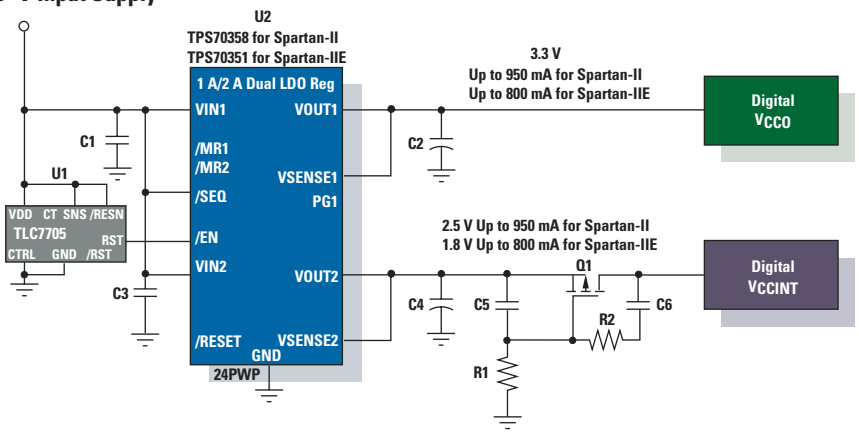
- **Modules**
  - Easiest switching DC/DC solution available. The module solution is complete, needing only the addition of an input and output capacitor for stability.
  - Double-sided modules save board space.
  - Full environmental qualification and EMI reports are available.

Spartan™-II/ Spartan™-IIE Design 1

Dual-Channel Low Dropout (LDO) Linear Regulator-Based, 800 mA V_{CCINT} Solution for Spartan-IIE, 950 mA V_{CCINT} Solution for Spartan-II

- Dual channel LDO in PowerPAD™ package saves cost and space.
- U1 monitors the input rail to ensure that it is up and stable before enabling the regulator.
- Linear regulators start-up fast. Q1 is used to provide soft start to V_{CCINT}.
- Sequencing V_{CCINT}, then V_{CCO} minimizes demand on the input supply.
- U2 is limited to 2 W @ T_A = 55 C and no airflow due to power dissipation. Its maximum output current is split between the rails.

5 -V Input Supply

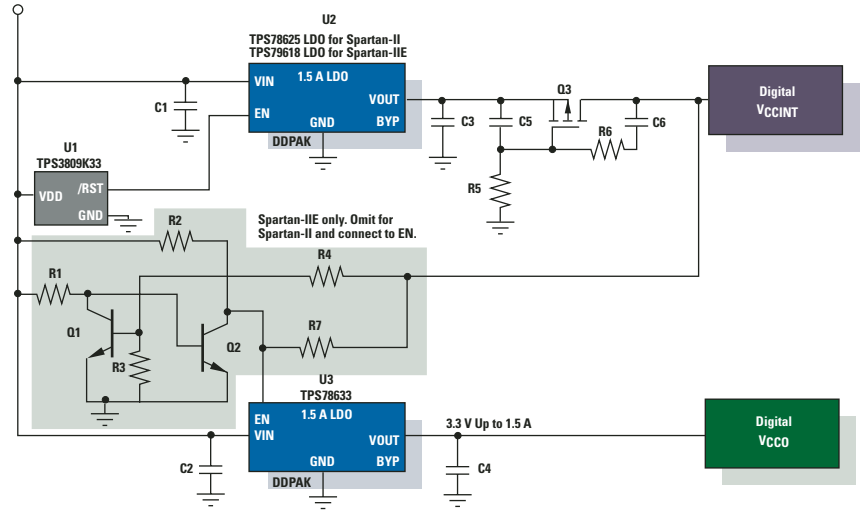


Spartan™-II/ Spartan™-IIE Design 2

Single-Channel LDO-Based, 1-A V_{CCINT} Solution for Spartan-IIE, 1.2 A V_{CCINT} Solution for Spartan-II

- Independent linear regulators allow higher power dissipation than an integrated dual-channel solution.
- Linear regulator solution saves cost and space over switching DC/DC solution.
- U1 monitors the input rail to ensure that it is up and stable before enabling the regulator.
- Linear regulators start-up fast. Q3 is used to provide soft start to V_{CCINT}.
- Sequencing V_{CCINT}, then V_{CCO} minimizes demand on the input supply.
- U2 is current limited @ T_A = 55 C and no airflow, due to power dissipation.
- Reduce cost by using lower current LDOs from U2, U3.
- Adapt to 3.3 V input:
  - Omit U3
  - Replace TPS3809K33 with TPS3809L30
  - With the lower V_{IN} to U2, the LDO can now support higher currents.

5-V Supply/Adaptable to 3.3 V

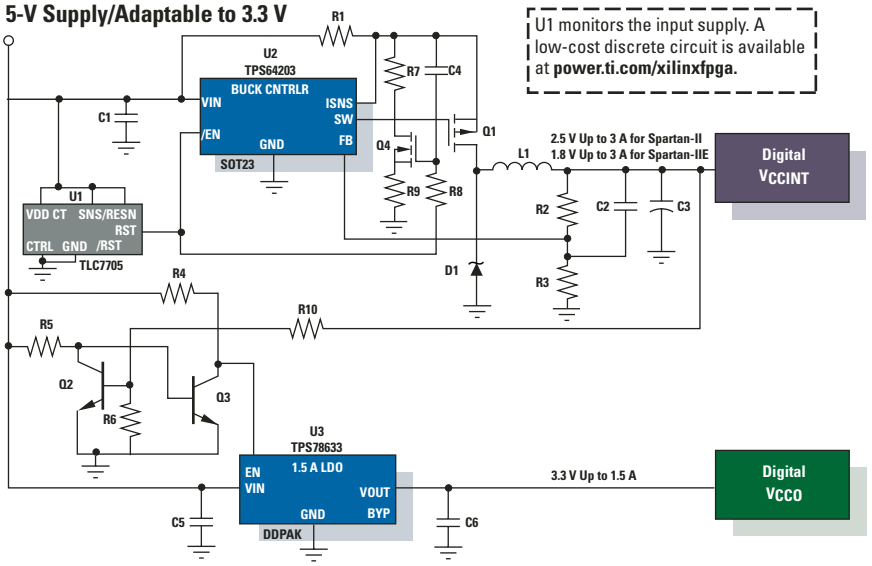


Spartan™-IIE Design 3*
Switching DC/DC Controller-Based, 3 A V_{CCINT} Solution

- Tiny SOT-23 switching DC/DC controller (U2) delivers up to 3 A at ultra-low cost.
- V_{CCINT} soft starts from internal soft start of U2.
- Sequencing of V_{CCINT} then V_{CCO} minimizes demand on the power supply.
- Size Q1 for the appropriate amount of current up to 3 A.
- Current sense resistor R1 gives more precise current limit; omit and connect ISNS to drain of Q1 to save cost and space.
- Reduce cost by using lower current LDOs from the TPS79xxx family for U3.
- Adapt to 3.3 V supply:
  - Omit U3 circuit
  - Replace TLC7705 with TLC7733

For Spartan-II, see **power.ti.com/xilinxfpga*

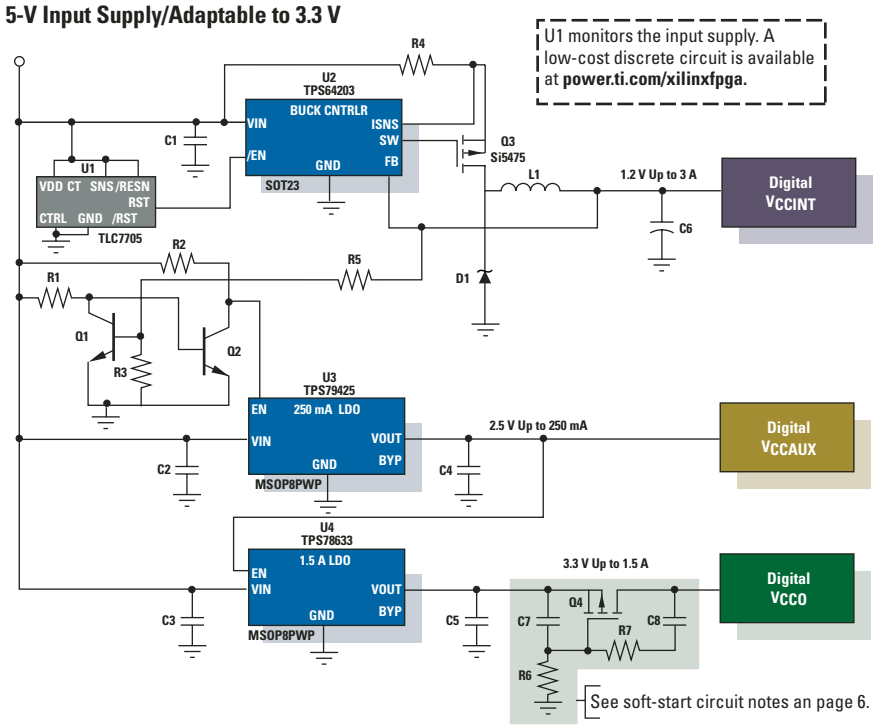
5-V Supply/Adaptable to 3.3 V



Spartan™-3 Design 1
Switching DC/DC Controller-Based, 3-A V_{CCINT} Solution

- Tiny SOT-23 switching DC/DC controller (U2) delivers up to 3 A at ultra-low cost.
- Minimum voltage ramp times are met by soft-starting. V_{CCINT} soft starts from internal soft start of U2. V_{CCO} soft starts using Q4 circuitry with U4.
- Sequencing of V_{CCINT} then V_{CCAUX} then V_{CCO} minimizes demand on the power supply.
- Size Q3 for the appropriate amount of current up to 3 A.
- Current sense resistor R4 gives more precise current limit; Omit and connect ISNS to drain of Q3 to save cost and space.
- Reduce cost by using lower current LDOs from the TPS79xxx family for U3, U4.
- Adapt for 3.3 V input supply:
  - Omit U4 circuit
  - Replace TLC7705 with TLC7733

5-V Input Supply/Adaptable to 3.3 V



See soft-start circuit notes an page 6.

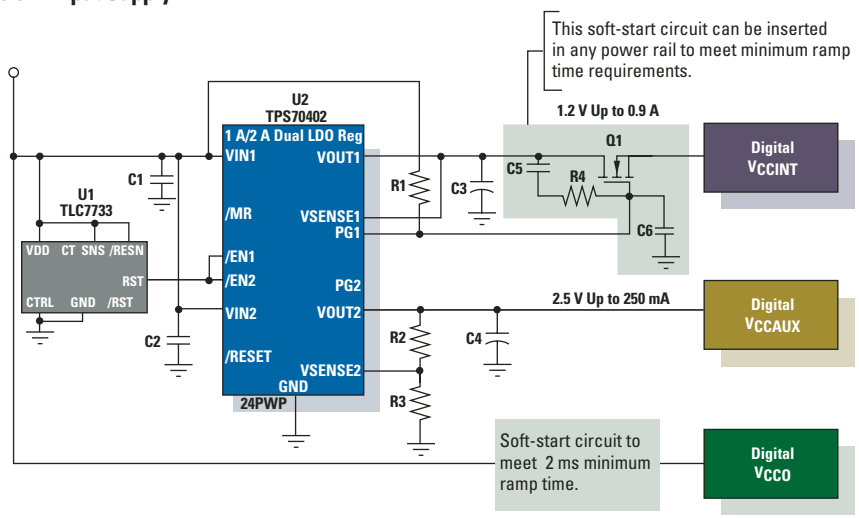
3.3 V Configuration

The Spartan-3 configuration and JTAG ports commonly use signals with a 2.5 V-swing. Alternatively, it is possible to use 3.3 V signals simply by adding a few external resistors. The 3.3 V signals can cause a reverse current that flows from certain configuration and JTAG input pins, through the FPGA, to the V_{CCAUX} power rail. It is necessary to limit this current to 10 mA (or less) per pin, and to manage the reverse current flowing out of the V_{CCAUX} pins. Please see the web for solution recommendations.

Spartan™-3/ Design 2
Dual LDO-Based, 900 mA V_{CCINT} Solution

- Dual channel LDO in PowerPAD™ package saves cost and space.
- U1 monitors the input rail to make sure that it is up and stable before enabling the regulators.
- Linear regulators startup fast. Q1 provides soft start to V_{CCINT} to meet ramp time requirements.
- Sequencing V_{CCINT} then V_{CCAUX} minimizes demand on the power supply.
- U2 is limited to 2 W @ T_A = 55 C and no airflow, due to power dissipation.

3.3-V Input Supply

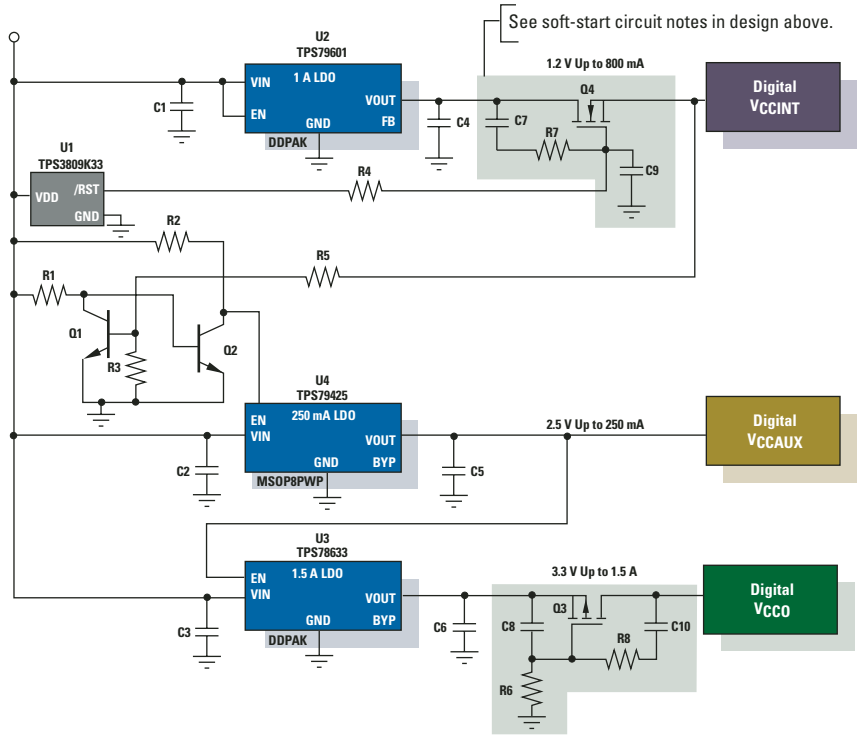


For 3.3 V configuration, see page 5.

Spartan™-3 Design 3
Single-Channel LDO-Based, 800 mA to 1.4 A V_{CCINT} Solution

- Independent linear regulators allow higher power dissipation than an integrated dual-channel solution.
- Linear regulator solution saves cost and space over a switching DC/DC solution.
- U1 monitors the input rail to make sure that it is up and stable before enabling the soft start circuit.
- Linear regulators startup fast. Q3 and Q4 are used to soft start the LDOs to meet ramp time requirements.
- Sequencing V_{CCINT} then V_{CCAUX} then V_{CCO} minimizes demand on the input supply.
- Reduce cost by using lower current LDOs from the TPS79xxx family for U2, U3, U4.
- Adapt for 3.3 V input supply:
  - Omit U3 circuit
  - U2 is shown current limited to 800 mA @ T_A = 55 C given a 5 V input supply and no airflow, due to power dissipation. With a 3.3 V supply, U2 will support up to 1.4 A if changed to a TPS78601 1.5 A LDO.
  - Replace TPS3809K33 with TPS3809L30
  - Resize R4

5-V Supply/Adaptable to 3.3 V

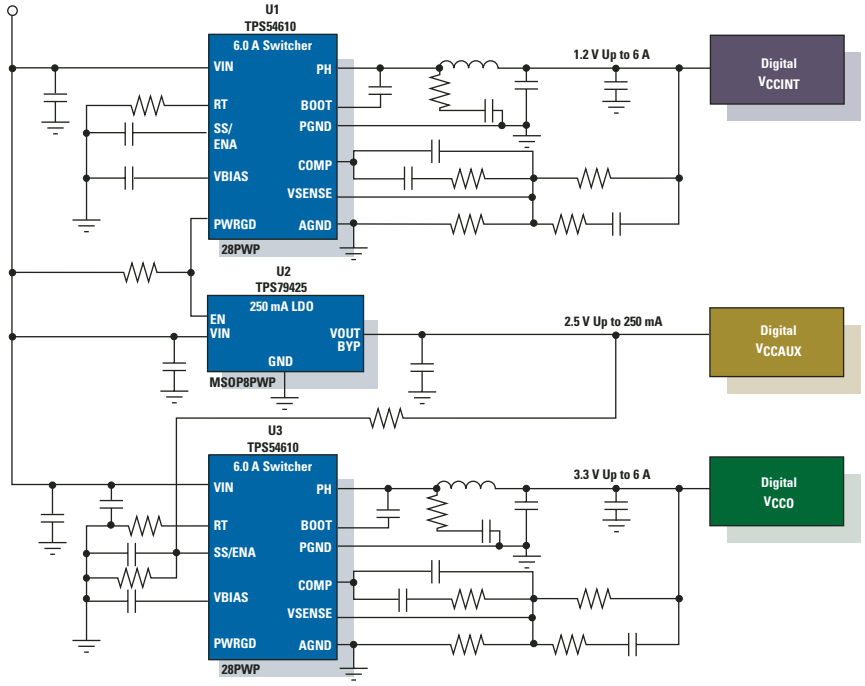


For 3.3 V configuration, see page 5.

Spartan™-3 Design 4
Switching DC/DC Converter-Based, 6-A V_{CCINT} Solution

- Highly efficient V_{CCINT} and V_{CCO} rails up to 6 A. Interchange any of the TPS54x10 family of parts for different current levels from 1.5 A to 9 A. *(See page 10 for pin compatibility).*
- SWIFT™ (Switcher with Integrated FET) TPS54610 adjustable design allows use of smaller inductor, ceramic capacitors.
- Fixed TPS5461x 1.2-V and 3.3-V design with internal compensation and tantalum capacitors available.
- Use SWIFT design software to customize external components, or see the web for complete schematics.
- High UVLO (Under Voltage Lockout) and integrated soft start of U1/U3 eliminate the need for an external SVS to monitor the input voltage. Soft starting also meets ramp time requirements.
- Sequencing of V_{CCO} then V_{CCAUX} then V_{CCO} minimizes demand on the power supply.
- Additional V_{CCO} rails easily added and sequenced using TPS54xxx PWRGD feature and enable.
- Adapt for 3.3 V supply:
  - Omit U3 circuit

5-V Input Supply/Adaptable to 3.3 V

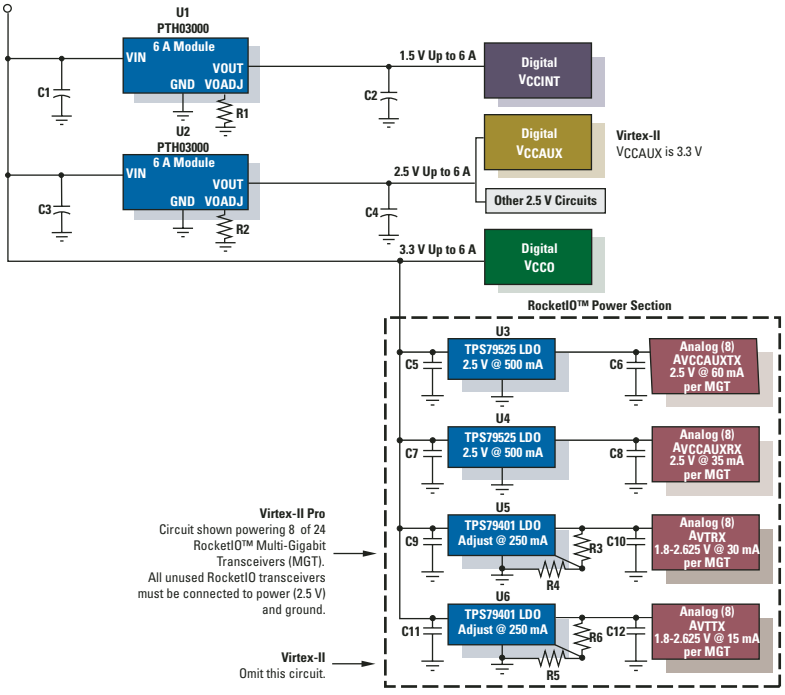


For 3.3 V configuration, see page 5.

Virtex™-II/ Virtex-II Pro™ Design 1
Module-Based, 6-A V_{CCINT} Solution

- Simple to use plug-in modules.
- Highly efficient V_{CCINT} and V_{CCO} rails.
- Additional V_{CCO} rails easily added.
- Interchange modules to support:
  - 6 A to 30 A
  - 5.0 V or 12 V input supply
  - Minimum input and output capacitors may change depending on application
- Reduce cost by using lower current LDOs from the TPS79xxx family for the RocketIO™ power section.

3.3 V Input Supply



RocketIO™ Power Section

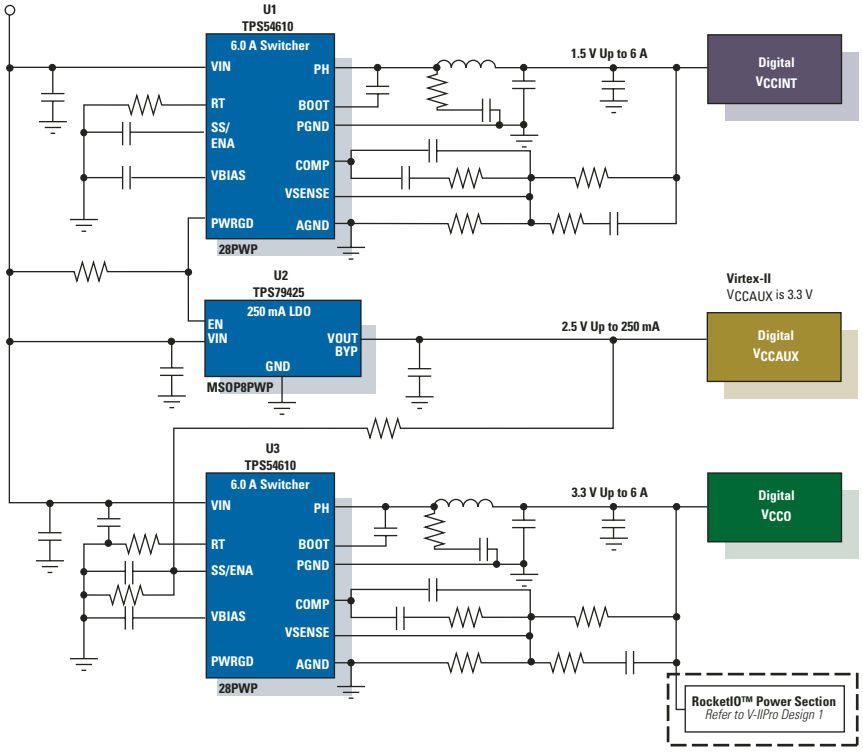
$AV_{CCAUXTX}$, $V_{CCAUXRX}$, AV_{TRX}, and AV_{TTX} are shown powered by independent linear regulators but they may all be powered by the same linear regulator, if desired. See TPS79xxx and TPS786xx LDO selection guide (see page 10).

Complete schematics available at: **power.ti.com/xilinxfpga**

Virtex™-II/ Virtex-II Pro™ Design 2
Switching DC/DC Converter-Based 6-A V_{CCINT} Solution

- Highly efficient V_{CCINT} and V_{CCO} rails up to 6 A. Interchange any of the TPS54x10 family of parts for different current levels from 1.5 A to 9 A. *(See page 10 for pin compatibility).*
- SWIFT™ (Switcher with Integrated FET) TPS54610 adjustable design allows use of smaller inductor, ceramic caps.
- Fixed TPS5461x 1.5-V and 3.3-V design with internal compensation and tantalum caps available.
- Use SWIFT design software to customize external components, or see the web for complete schematics.
- High UVLO and integrated soft start of U1/U3 eliminate the need for an external SVS to monitor the input voltage.
- Sequencing of V_{CCINT} then V_{CCAUX} then V_{CCO} minimizes demand on the input supply.
- Additional V_{CCO} rails easily added and sequenced using TPS54xxx PWRGD feature and enable.
- Adapt for 3.3 V supply:
  - Omit U3 circuit

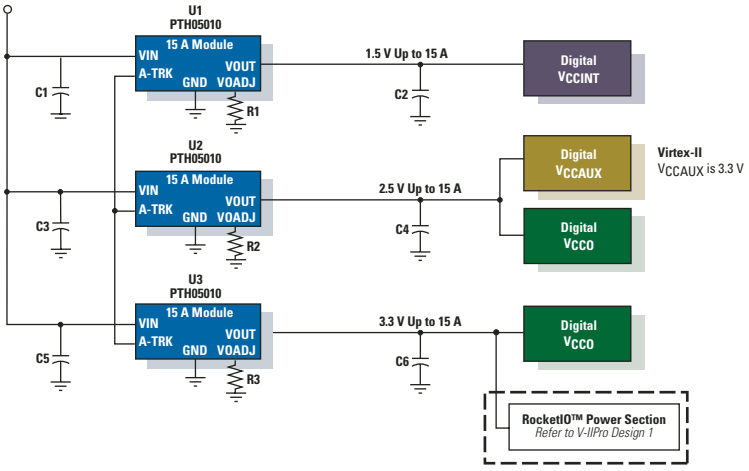
5-V Supply/Adaptable to 3.3 V



Virtex™-II/ Virtex-II Pro™ Design 3
Modular-Based 15 A V_{CCINT} Solution

- Simple to use plug-in modules.
- Powers one or multiple FPGAs.
- Highly efficient V_{CCINT} and V_{CCO} rails.
- High UVLO and soft start of U1/U2/U3 eliminate the need for an external SVS to monitor the input voltage.
- Auto-Track™ connected as shown provides simultaneous sequencing. Supply voltage supervisors can be added to sequentially sequence to reduce demand on the power supply. See the web for alternate implementations.
- Interchange modules to support:
  - 6 A to 30 A
  - 3.3 V or 12 V input supply
  - Minimum input and output capacitors may change depending on application

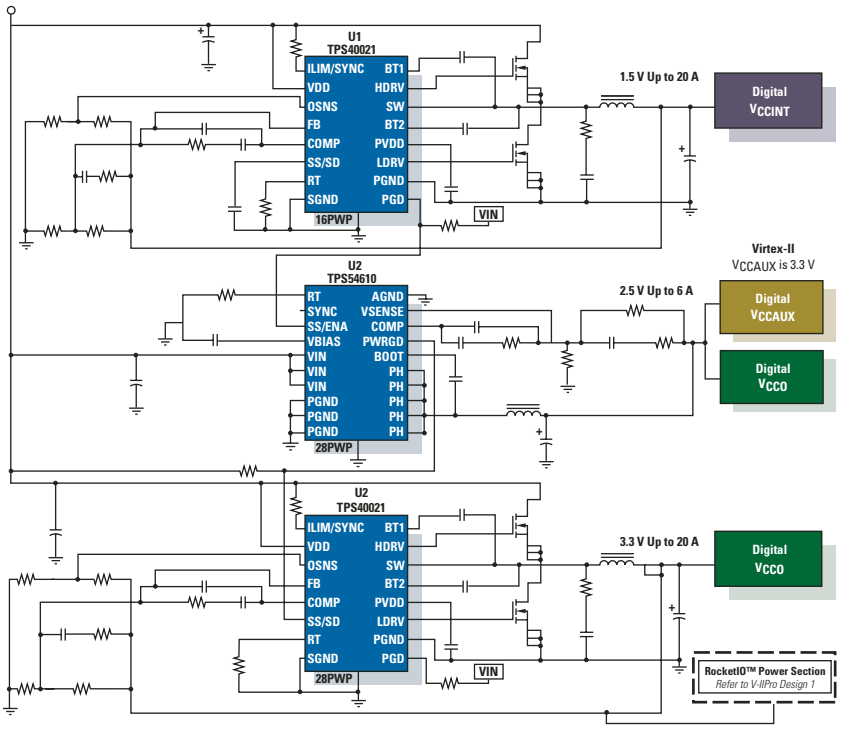
5-V Supply/Adaptable to 3.3 V or 12 V



Virtex™-II/ Virtex-II Pro™ Design 4
Low-Voltage Switching DC/DC Controller-Based, 20-A V_{CCINT} Solution

- Powers one or multiple FPGAs.
- Highly efficient V_{CCINT} and V_{CCO} rails up to 20 A.
- Flexible controller design allows optimization for size, power dissipation, and cost.
- Use TPS40K and SWIFT™ design software to customize designs, or see the web for complete schematics.
- Adjustable TPS541x allows use of smaller inductor, ceramic capacitors.
- Fixed TPS5461x design with internal compensation and tantalum caps available.
- Interchange any of the TPS54x10 SWIFT™ family of parts for different current levels from 1.5 A to 9 A. *(See page 10 for pin-pin compatibility).*
- Soft-start feature of U1, U2, and U3 eliminates the need for an external SVS to monitor the input voltage.
- Sequencing minimizes demand on the input supply.
- Additional V_{CCO} rails easily added and sequenced.
- Adapt for 3.3 V supply:
  - Omit U3 circuit

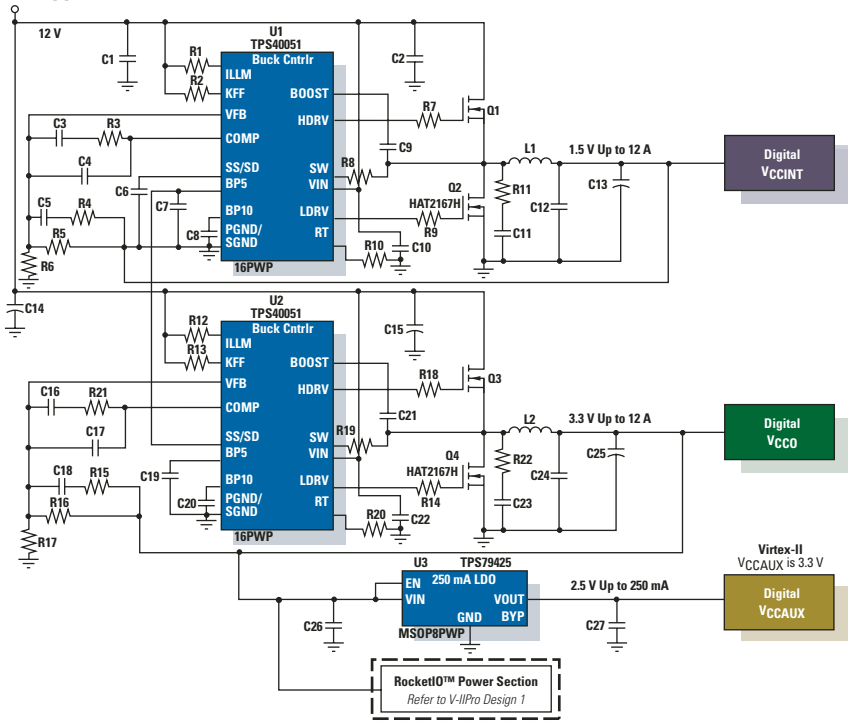
5-V Supply/Adaptable to 3.3 V



Virtex™-II/ Virtex-II Pro™ Design 5
High-Voltage Switching DC/DC Controller-Based, 12-A V_{CCINT} Solution

- Powers one or multiple FPGAs.
- Highly efficient V_{CCINT} and V_{CCO} rails up to 12 A.
- Flexible controller design allows optimization for size, power dissipation, and cost.
- Use TPS40K design software to customize designs, or see the web for complete schematics.
- Soft-start feature and high UVLO of U1/U2 eliminates the need for an external SVS to monitor the input voltage.
- Sequencing minimizes demand on the input supply.
- Additional V_{CCO} rails easily added and sequenced.
- Reduce cost by using lower current LDOs from the TPS79xxx family for U3.

12-V Supply



Complete schematics available at: **power.ti.com/xilinxfpga**

Selection Guides

Low Dropout Regulators (LDO)

Device[1]	I_O (mA)	V_{DO} @ I_O (mV)	I_q (µA)	Output Options: Fixed Voltage (V) (V)	Output Options: Adj.	Min V_{IN}	Max V_{IN}	Accuracy (%)	Packages: SOT23	Packages: MSOP	Packages: SOT223	Packages: DDPAK	Features[2]	C_O[3]	Comments	Price[4]
Positive Voltage, Single Output Devices																
TPS793xx	200	77	170	1.8, 2.5, 2.8, 2.85, 3.0, 3.3, 4.75	1.2 to 5.5	2.7	5.5	2	✔				EN, BP	2.2 µF C	RF Low Noise, High PSRR	0.40
TPS794xx	250	145	170	1.8, 2.5, 2.8, 3.0, 3.3	1.2 to 5.5	2.7	5.5	3		✔	✔		EN, BP	2.2 µF C	RF Low Noise, High PSRR	0.65
TPS795xx	500	105	265	1.6, 1.8, 2.5, 3.0, 3.3	1.2 to 5.5	2.7	5.5	2			✔		EN, BP	1 µF C	RF Low Noise, High PSRR	0.95
TPS796xx	1000	200	265	1.8, 2.5, 2.8, 3.0, 3.3	1.2 to 5.5	2.7	5.5	2			✔	✔	EN, BP	1 µF C	RF Low Noise, High PSRR	1.05
TPS786xx	1500	390	265	1.8, 2.5, 2.8, 3.0, 3.3	1.2 to 5.5	2.7	5.5	2			✔	✔	EN, BP	1 µF C	RF Low Noise, High PSRR	1.30
UC382-x	3000	350	6 mA	1.5, 2.1, 2.5	1.20 to 6.0	1.7	7.5	1				✔	—	100 µF T	Separate V_{BIAS}	2.60

[1] *xx represents the voltage option. For example, 33 represents the 3.3-V option. The adjustable output voltage option is represented by 01.*

[2] *EN = Active High Enable, BP = Bypass Pin for noise reduction capacitor.*

[3] *T= Tantalum, C=Ceramic output capacitor. TPS795/796/786xx minimum C_O is only 1 uF but 2.2 uF was used in the reference designs to minimize the bill of materials since the input capacitor requirement is 2.2 uF.*

[4] *Suggested resale price in U.S. dollars in quantities of 1,000.*

Dual Output LDOs

Device	I_{O1} (mA)	I_{O2} (mA)	V_{DO1} @ I_{O1} (mV)	V_{DO2} @ I_{O2} (mV)	I_q (µA)	Output Options: Fixed Voltage (V)	Output Options: Adj.	Accuracy (%)	PWP Package	Min V_O	Max V_O	Features: /EN	Features: PG	Features: SVS	Features: Seq	Features: Low Noise	Features: Min V_{IN}	Features: Max V_{IN}	Features: C_O[1]	Description	Price[2]
TPS703xx	2000	1000	160	—	185	3.3/2.5, 3.3/1.8, 3.3/1.5, 3.3/1.2	✔	2	✔	1.2	5.5	✔	✔	✔	✔	✔	2.7	5.5	22 µF T	Dual Output LDO with Sequencing	2.30
TPS704xx	2000	1000	160	—	185	3.3/2.5, 3.3/1.8, 3.3/1.5, 3.3/1.2	✔	2	✔	1.2	5.5	✔	✔	✔		✔	2.7	5.5	22 µF T	Dual Output LDO with Independent Enable	2.30

[1] *T = Tantalum output capacitor.*

[2] *Suggested resale price in U.S. dollars in quantities of 1,000*

Switching DC/DC Converters

Device	I_{OUT} (mA)	V_{IN} (V)	Adj (V)	V_{OUT} Fix (V)	V_{OUT} Efficiency %	Switching Frequency (max) (kHz)	Quiescent Current (typ) (mA)	Features: Shutdown	Features: Power Good	Features: Dual Input Bus (3.3, 2.5 V)	Features: Current Limit	Packaging: Thermal Limit	Packaging: TSSOP	EVM	Price[1]
SWIFT™ Step Down (Buck) Converters — up to 9 A															
TPS54110	1500	3.0 to 6.0	0.9 to 4.5	—	90	700	3.6	✔	✔		✔	✔	20	✔	2.15
TPS54310	3000	3.0 to 6.0	0.9 to 4.5	—	90	700	6.2	✔	✔		✔	✔	20	✔	2.15
TPS54311	3000	3.0 to 6.0	—	0.9	90	700	6.2	✔	✔		✔	✔	20		2.15
TPS54312	3000	3.0 to 6.0	—	1.2	90	700	6.2	✔	✔		✔	✔	20		2.15
TPS54313	~3000	3.0 to 6.0	—	1.5	90	700	6.2	✔	✔		✔	✔	20		2.15
TPS54314	3000	3.0 to 6.0	—	1.8	90	700	6.2	✔	✔		✔	✔	20	✔	2.15
TPS54315	3000	3.0 to 6.0	—	2.5	90	700	6.2	✔	✔		✔	✔	20		2.15
TPS54316	3000	3.0 to 6.0	—	3.3	90	700	6.2	✔	✔		✔	✔	20		2.15
TPS54350	3000	4.5 to 20	0.9 to 12	—	85	700	9	✔	✔		✔	✔	16	✔	2.35
TPS54610	6000	3.0 to 6.0	0.9 to 4.5	—	90	700	11	✔	✔		✔	✔	28	✔	4.10
TPS54611	6000	3.0 to 6.0	—	0.9	90	700	11	✔	✔		✔	✔	28		4.10
TPS54612	6000	3.0 to 6.0	—	1.2	90	700	11	✔	✔		✔	✔	28		4.10
TPS54613	6000	3.0 to 6.0	—	1.5	90	700	11	✔	✔		✔	✔	28		4.10
TPS54614	6000	3.0 to 6.0	—	1.8	90	700	11	✔	✔		✔	✔	28	✔	4.10
TPS54615	6000	3.0 to 6.0	—	2.5	90	700	11	✔	✔		✔	✔	28		4.10
TPS54616	6000	3.0 to 6.0	—	3.3	90	700	11	✔	✔		✔	✔	28		4.10
TPS54810	8000	4.0 to 6.0	0.9 to 4.5	—	85	700	11	✔	✔		✔	✔	28	✔	4.20
TPS54910	9000	3.0 to 4.0	0.9 to 2.5	—	90	700	11	✔	✔		✔	✔	28	✔	4.40
TPS54974	9000	2.2 to 4.0	0.2 to 2.5	—	90	700	11	✔	✔	✔	✔	✔	28	✔	4.40

[1] *Suggested resale price in U.S. dollars in quantities of 1,000.*

New products appear in **BOLD RED**.

Selection Guides

Switching DC/DC Controllers

Device	V_{IN} (V)	V_O (max) (V)	V_O (min) (V)	V_{ref} Tol (%)	Driver Current (A)	Output Current Range (A)	Multiple Outputs	Adaptive Voltage Positioning	Protection[1]	Comments	Price[2]
Performance Processor Power Supply Controllers (Synchronous Rectification)											
TPS64200	1.8 to 6.5	6.5	1.2	2	0.150	0 to 3	No	No	OCP, UVLO	Non-sync buck in SOT-23	0.50
TPS40001	2.25 to 5.5	4	0.7	1.5	1	3 to10	No	No	OCP, UVLO	300-kHz switching frequency gives highest efficiency for lowest heat	0.99
TPS40003	2.25 to 5.5	4	0.7	1.5	1	3 to10	No	No	OCP, UVLO	600-kHz switching frequency gives smallest solution	0.99
TPS40021	2.25 to 5.5	4	0.7	1	1	10 to 20	No	No	OCP, UVLO	Enhanced flexibility with user programmability	1.15
TPS40051	8 to 40	30	0.7	1	1	3 to15	No	No	OCP, UVLO	Wide input range sync buck, source/sink	1.35
TPS40061	10 to 55	40	0.7	1	1	1 to 8	No	No	OCP, UVLO	Wide input range sync buck, source/sink	1.35

[1] *OCP — over-current protection; UVLO — under-voltage lockout.*

[2] *Suggested resale price in U.S. dollars in quantities of 1,000.*

New products appear in **BOLD RED**.

Plug-In Power Solutions

Device[1]	Input Bus Voltage	P_{OUT} or I_{OUT}	Isolated Outputs	V_O Range (V)	V_O Adjustable	Price[2]
Non-Isolated Single Positive Output						
PTH03010/20/30/50/60	3.3 V	15 A, 22 A, 30 A, 6 A, 10 A	No	0.8 to 2.5	Yes	14.00, 18.15, 24.65, 9.10, 11.20
PTH05010/20/30/50/60	5 V	15 A, 22 A, 30 A, 6 A, 10 A	No	0.8 to 3.6	Yes	14.00, 18.15, 24.65, 9.10, 11.20
PTH12010/20/30/50/60	12 V	12 A. 18 A, 26 A, 6 A, 8 A	No	1.2 to 5.5	Yes	14.00, 18.15, 24.65, 9.10, 11.20

[1] *See* **power.ti.com** *for a complete product offering.*

[2] *Suggested resale price in U.S. dollars in quantities of 1,000.*

New products appear in **BOLD RED**.

Supply Voltage Supervisors (SVS)

Device	Number of Supervisors	Supervised Voltages	Packages	I_{DD} (typ) (µA)	Time Delay (ms)	Watchdog Timer WDI (sec)	Manual Reset Input/MR	Active-High Reset Output	Reset Output Topology[1]	Comments	Price[2]
TPS3809	1	2.5/3.0/3.3/5.0	SOT-23	9	200	—			PP[1]	Small, low cost	0.29
TLC77xx	1	Adj./2.5/3.3/3.0/5.0	SO-8, DIP-8, TSSOP-8	9	Prog	—		✔	PP	Universal SVS with broad voltage range and both active-low and active-high reset	0.60

[1] *PP = Push-Pull.*

[2] *Suggested resale price in U.S. dollars in quantities of 1,000.*

Active Bus Terminator* (DDR) Converter (with Integrated FETs) Solutions

Device	I_{OUT} (mA)	V_{IN} (V)	Adj. (V)	V_{OUT} Efficiency %	(max) (kHz)	Switching Frequency TSSOP	EVM	Price[1]
TPS54372	3000	3.0 to 6.0	0.2 to 4.5	90	700	20	✔	3.05
TPS54672	8000	3.0 to 6.0	0.2 to 4.5	90	700	28	✔	4.10
TPS54872	8000	4.0 to 6.0	0.2 to 4.5	85	700	28	✔	4.20
TPS54972	9000	3.0 to 4.0	0.2 to 4.5	90	700	28	✔	4.40

**Tracks externally applied reference voltage.*

[1] *Suggested resale price in U.S. dollars in quantities of 1,000.*

Active Bus Terminator* (DDR) Controller (with External FETs) Solutions

Device	I_{OUT1} (V_{DDQ}) (A)	I_{OUT2} (V_{TT}) (A)	I_{OUT3} (buf. V_{REF}) (mA)	V_{IN} (V)	V_{OUT1} (V_{DDQ}) Adj. (V)	V_{OUT2} (V_{TT}) Fixed (V)	V_{OUT3} (buf. V_{ref}) Fixed (V)	Switching Frequency Selectable (kHz)	Light Load Efficiency Mode	Selectable Output Discharge	TSSOP	Price[1]
TPS51020	10	5	3	4.5 V to 28 V	0.9 to 90% V_{IN}	1/2 V_{DDQ}	1/2 V_{DDQ}	270, 360, 450	Yes	Yes	30	3.00

**Solution includes V_{DDQ}, V_{TT}, and buffered V_{REF} outputs.*

[1] *Suggested resale price in U.S. dollars in quantities of 1,000.*

New products appear in **BOLD RED**.

TI's PTH Series Modules offer efficiency up to 96% and power densities up to 235 W/in^3. Please refer to pages 7 and 8 for power solutions using these modules.

TI Worldwide Technical Support

Internet

TI Semiconductor Product Information Center Home Page
support.ti.com

TI Semiconductor KnowledgeBase Home Page
support.ti.com/sc/knowledgebase

Product Information Centers

Americas

Phone	+1(972) 644-5580
Fax	+1(972) 927-6377
Internet/Email	support.ti.com/sc/pic/americas.htm

Europe, Middle East, and Africa

Phone	
Belgium (English)	+32 (0) 27 45 55 32
Finland (English)	+358 (0) 9 25173948
France	+33 (0) 1 30 70 11 64
Germany	+49 (0) 8161 80 33 11
Israel (English)	1800 949 0107
Italy	800 79 11 37
Netherlands (English)	+31 (0) 546 87 95 45
Spain	+34 902 35 40 28
Sweden (English)	+46 (0) 8587 555 22
United Kingdom	+44 (0) 1604 66 33 99
Fax	+(49) (0) 8161 80 2045
Internet	support.ti.com/sc/pic/euro.htm

Japan

Fax	International	+81-3-3344-5317
	Domestic	0120-81-0036
Internet/Email	International	support.ti.com/sc/pic/japan.htm
	Domestic	www.tij.co.jp/pic

Asia

Phone	
International	+886-2-23786800
Domestic	Toll-Free Number
Australia	1-800-999-084
China	800-820-8682
Hong Kong	800-96-5941
Indonesia	001-803-8861-1006
Korea	080-551-2804
Malaysia	1-800-80-3973
New Zealand	0800-446-934
Philippines	1-800-765-7404
Singapore	800-886-1028
Taiwan	0800-006800
Thailand	001-800-886-0010
Fax	886-2-2378-6808
Email	tiasia@ti.com
	ti-china@ti.com
Internet	support.ti.com/sc/pic/asia.htm

B111103

Texas Instruments

SLPB008

PCI

AWESOME ASIC Prototyping Solutions

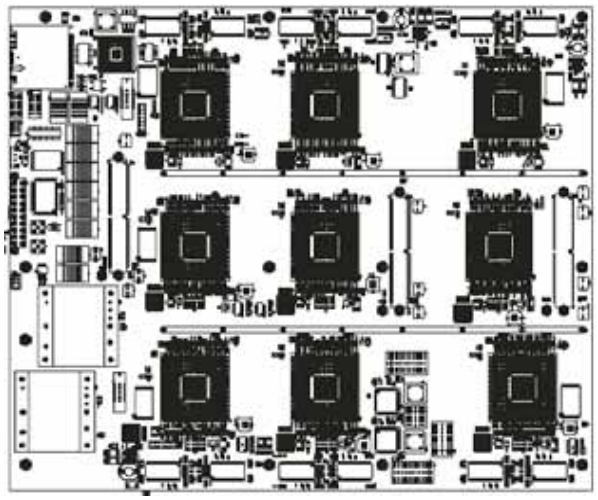
PCI-X

Cool Emulation Platforms

DN6000K10SC

Single VirtexII - Pro PWB

- 2VP20/30/40/50 (all speed grades)
- 2 Separate DDR SDRAM's (16Mb x 16)
- 2 Separate SSRAM's (512k x 36)
- 1 Flash's (4M x 16)
- Misc. serial I/O's -- via SMA's



DN6000K10S

Single VirtexII - Pro PWB

- 2VP70/2VP100 (all speed grades)
- 4 Separate DDR SDRAM's (16Mb x 16)
- 4 Separate SSRAM's (512k x 36)
- 2 Flash's (4M x 16)
- Misc. serial I/O's
- 2 PowerPC's
- PCI / PCI-X or standalone

DN3000K10S

Single VirtexII™ PWB

- 2V4000/6000/8000
- 2 PLL's for high speed clock distribution
- 3 512k x 36 SSRAM's
- 1 GB SDRAM DIMM (PC133)
- FPGA configuration using SmartMedia Card
- Sun, Win2000/NT/98, LINUX Drivers and Utilities

DN6000k10 - The MONSTER

Up to 9 VirtexII-Pro FPGAs

- 2vp70/2vp100 (FF1704)
- 12 DDR SDRAM's (16/32M x16)
- 5 FLASH (4M x 16)
- USB Hosted
- 18 PowerPC's

DN3000K10

Logic Emulation PWB with the Xilinx VirtexII™ FPGA's.

- Up to 5 2V6000/2V8000's per PWB
  -- 466,000 FF's!
- 2 PLL's for high speed clock distribution
- 4 512k x 36 SSRAM's
- 1 GB SDRAM DIMM (PC133)
- Monstrous amounts of interconnect
- FPGA configuration using SmartMedia Card

Consulting Services:

- ASIC & FPGA Design Translation
- HW Emulation and Testing
- Algorithmic Acceleration
- FPGA/ASIC/Board Design
- Turnkey Verilog/VHDL

Rock Star* the kitty says:

"This stuff is WAY cool!"

- Persian Hunting Cat

Mr. Lazy* Says:

"I hate searching for bugs; these guys made my life easy!"

- Blue Naped Mousebird

The DINI Group

1010 Pearl Street, Suite 6
La Jolla, CA 92037-5165
http://www.dinigroup.com
Voice: (858) 454-3419
FAX: (858) 454-1728
Email: sales@dinigroup.com

The DiNI Group

Xilinx Partner Yellow Pages

Xilinx Alliance EDA Members and Products

AccelChip
www.accelchip.com
High-level synthesis tools that read MATLAB and output RTL

Alatek
www.alatek.com
Hardware acceleration (HES)

Altium
www.altium.com/products/nvisage.htm
Complete PCB/FPGA design flow (nVisage) targeting Spartan™ Series FPGAs

Ansoft
www.ansoft.com
High-Frequency products for system analysis, circuit design, and electromagnetic simulation (Ansoft Designer™, HFSS™)

Apache Design
www.apache-da.com/Products/nspice.htm
HSPICE®-compatible simulator using actual S-parameter data (NSPICE) for Virtex-II Pro™ MGT-based PCB design

Aptix
www.aptix.com
ASIC emulation (Prototype Studio™)

Atrenta
www.atrenta.com
RTL Linter technology for Virtex™-II and Virtex-II Pro FPGAs (SpyGlass®)

Auspy
www.auspy.com
FPGA partition (APS II) and ASIC emulation (ACE Compiler)

Cadence Design Systems
www.cadence.com/feature/fpga_design.html
Virtex-II Pro MGT support for high-speed PCB design (SPECCTRAQuest®); NCSim family for FPGA design simulation

Celoxica
www.celoxica.com
C-level design creation, compilation, and verification for Virtex-II and Virtex-II Pro FPGAs

Endeavor
www.endeav.com
Co-verification for Virtex-II Pro (CoSimple™)

EVE
www.eve-team.com
RTL-level Simulator Accelerator (ZEBU)

Forte Design
www.forteds.com
HLL compiler to RTL (Cynlib Tool Suite); timing specification and analysis tool (TimingDesigner)

Hier Design Inc.
www.hierdesign.com
Hierarchical floorplanning and analysis software (PlanAhead™) for Virtex-II, Virtex-II Pro and Spartan-3 FPGAs

MAGMA
www.magma-da.com
Architecture-specific synthesis and layout optimization for Virtex-II series FPGA (PALACE™)

The MathWorks
www.mathworks.com
System-level design tools for FPGA and processor-based signal processing systems

Mentor Graphics
www.mentor.com
Virtex-II Pro MGT support for high-speed PCB design (HyperLynx™, ICX™, Tau™); complete FPGA design flow using the ModelSim® family of simulators and Leonardo-Spectrum™/Precision® synthesis; Seamless® co-verification support for Virtex-II Pro; FPGA BoardLink™ FPGA symbol generation for fast PCB design iterations for Virtex-II and Virtex-II Pro FPGAs

Novilit
www.novilit.com
Design partitioning using ISE EDK(AnyWare) for Virtex-II and Virtex-II Pro FPGAs

Product Acceleration Inc.
www.prodacc.com
FPGA symbol generation (LiveComponent™); automated pin assignment optimization; PCB layer count control (DesignF/X)

Simucad
www.simucad.com
Verilog™ simulator for FPGAs (Silos)

Synopsys
www.synopsys.com
Virtex-II Pro multigigabit transceiver support for high-speed PCB design (HSPICE®); FPGA Compiler II™ RTL synthesis for Virtex series FPGAs; VCS™(MX)/Scirocco™(MX) simulators; Formality® formal verification, LEDA® RTL linter, and PrimeTime™ static timing analysis; high-level design using CoCentric System Studio

Synplicity
www.synplicity.com
RTL synthesis (Synplify®), debugger technology (Identify®), physical synthesis (Amplify®), and ASIC prototyping (Certify®) for Xilinx FPGAs

Xilinx AllianceCORE Third-Party IP Providers and Products

Alcatel Internetworking, Inc.
www.ind.alcatel.com/emac/
Ethernet media access controller cores

Amphion Semiconductor Ltd.
www.amphion.com
DSP, wireless communications, and multimedia cores

Barco-Silex
www.barco-silex.com
DSP cores, with a specialization in image-audio applications

CAST, Inc.
www.cast-inc.com
General-purpose IP for processors, peripherals, multimedia, networking, encryption, serial communications, and bus interfaces

Crossbow Technologies, Inc.
www.crossbowip.com
System Interconnect IP for on-chip and chip-to-chip multiprocessing

Deltatec S.A.
www.deltatec.be/
IP and services for digital imaging, DSP, multimedia, PCI, and datacom

Derivation Systems, Inc.
www.derivation.com
IP cores for high assurance hardware/software applications and embedded systems products; 32-bit Java processor core

Digital Communications Technologie
www.dctl.com
Embedded Java solutions

Digital Core Design
www.dcd.pl/
Microprocessor, microcontroller, floating point, and serial communication controller cores

Dolphin Integration
www.dolphin.fr/
Bus interface, processor, peripheral, and DSP cores; EDA software

Duma Video, Inc.
www.dumavideo.com
Digital video, AES/EBU audio, MPEG and DCT cores; real-time MPEG-2 HD and SD compression

eInfochips Pvt. Ltd.
www.einfochips.com
PCI, USB, 80186, multimedia, and wireless cores

Eureka Technology
www.eurekatech.com
Silicon-proven IP for CPU interface, PCI, PCMCIA, memory control, and other peripheral functions for SoC designs

GDA Technologies, Inc.
www.gdatech.com
HyperTransport, PCI, Ethernet, design services

GV & Associates, Inc.
www.gvassociates.com
Boards and expertise in DSP, processor, software, analog, and RF design

Helion Technology Limited
www.heliontech.com
Encryption and DSP algorithms; design services

iCoding Technology, Inc.
www.icoding.com
Turbo Code-related error correction products including system design

Intrinsyc, Inc.
www.intrinsyc.com
Single board embedded computers (FPGA + external processor); peripheral IP

Loarant Corporation
www.loarant.com
Proprietary 16-bit RISC CPU; embedded system design

MEET Ltd.
www.meet-electronics.com
IP cores dedicated to industrial servomotor control

Memec Core
www.memecdesign.com/xilinx/
Peripheral, bus interfaces, encryption, and communications cores

ModelWare, Inc.
www.modelware.com
Datacom/telecom cores including ATM, HDLC, POS, IMA, UTOPIA, and bridges

NewLogic GmbH
www.newlogic.com
Cores for wireless applications including Bluetooth and 802.11

Paxonet Communications
www.paxonet.com
Solutions for interworking metro networking technologies to SONET

Perigee, LLC
www.perigeellc.com
DSP and image processing cores; embedded firmware, GUI, and application software development

Pinpoint Solutions, Inc.
www.asic-design.com
Standards-based telecom and datacom IP

QualCore Logic, Inc.
www.virtualipgroup.com
Cores for microcontrollers, telecommunications, datacom, PC peripherals, and bus interface standards

RealFast Operating Systems AB
www.realfast.se/
Real-time systems and RTOS; operating system cores

Robert Bosch GmbH
www.can.bosch.com
Automotive, CAN

SOC Solutions, L.L.C.
www.socsolutions.com
Embedded µP and software; networking, wireless, audio, GPS, and handheld; peripheral cores

SysOnChip, Inc.
www.sysonchip.co.kr/
IP and products for CDMA Cellular/PCS/WLL modem, FEC, and Bluetooth™

Telecom Italia Lab S.p.A.
www.idosoc.com/soc/viplibrary/
Cores for ATM, optical networking, and switched digital video broadcast equipment

Tensilica, Inc.
www.tensilica.com
Configurable processor cores and development tools

TurboConcept
www.turboconcept.com
Turbo Code forward error correction solutions

Wipro, Ltd.
www.wipro.com
IP building blocks for technologies like IEEE 1394, USB, Bluetooth, and IEEE 802.11

Xylon d.o.o.
www.logicbricks.com
Human-machine interfaces and industrial communication controllers

Zuken, Inc.
www.zuken.co.jp/soc/
PCI2.2, 10/100 Ethernet, and Gigabit Ethernet cores

Reference Design Partners

The Xilinx Reference Design Alliance Program builds partnerships with industry-leading semiconductor manufacturers to develop reference designs for accelerating our customer's product and system time to market. The goal is to build a library of high-quality, multicomponent system-level reference designs in the areas of networking, communications, image and video processing, DSP, and emerging technologies and markets.

Agere Systems
www.agere.com

AMCC
www.amcc.com

Analog Devices
www.analog.com

Bay MicroSystems
www.baymicrosystems.com

BitBlitz Communications
www.bitblitz.com

Broadcom Corp.
www.broadcom.com

Centillium
www.centillium.com

Cypress Semiconductor
www.cypress.com

IBM Microelectronics
www.ibm.com/us/

IDT
www.idt.com

Ignis Optics
www.ignis-optics.com

Infineon Technologies AG
www.infineontechnologies.com

Intel Corporation
developer.intel.com/design/network/

Intrinsity
www.intrinsity.com/home.htm

Micron Technology
www.micron.com

Motorola
www.motorola.com

National Semiconductor Corporation
www.national.com

NetLogic Microsystems, Inc.
www.netlogicmicro.com

Octasic Semiconductor
www.octasic.com

PMC-Sierra Inc.
www.pmc-sierra.com/index.html

Samsung Semiconductor, Inc.
www.usa.samsungsemi.com

SiberCore Technologies
www.sibercore.com

Texas Instruments
www.ti.com

TransSwitch Coporation
www.transwitch.com/index_flash.jsp

Vitesse
www.vitesse.com

ZettaCom
www.zettacom.com

XPERTS Partners

3T BV
www.3T.nl/
Design services; measurement instruments; fail-safe controlling; medical equipment, communication, and computer peripherals

Accent S.r.l.
www.Accent.it/
Complex FPGAs; SoCs; PowerPCs™; IP platforms; software, PCB, and IC foundry services, from concept to silicon

ADI Engineering
www.adiengineering.com
Reference designs; custom designs; design services focused on Intel IXA and XScale™

Advanced Electronic Designs, Inc.
www.aedbozeman.com
Design services and manufacturing support; hardware and software design for embedded systems

Advanced Principles Group, Inc.
www.advancedprinciples.com
Design services and consulting; software algorithm acceleration and application-specific computing systems

Alpha Data
www.alpha-data.com
PCI/PMC/CPCI FPGA boards; embedded applications using HDL or System Generator methodologies

AMIRIX Systems Inc.
www.amirix.com
Design services for electronic product realization; high-performance digital boards; programmable logic; embedded software

Andraka Consulting Group, Inc.
www.andraka.com
Exclusively FPGAs for DSP since 1994; applications include radar, sonar, digital communications, imaging, and video

Array Electronics
www.array-electronics.de/
SoC/FPGA/PCB designs in telecom, industrial, and DSPs; concepts; verification; prototypes; cores

Avnet Design Services
www.ads.avnet.com
Embedded processing FPGA design services; Virtex-II Pro; high-speed SERDES; IP core integration; timing closure; RLDRAM; RapidIO™

Baranti Group Inc.
www.baranti.com
Award-winning design/manufacturing firm; expertise in digital video, FPGAs

Barco Silex
www.barco-silex.com
Total solutions consisting of IP, FPGA, and board design

Benchmark Electronics Inc.
www.bench.com
EMS/JDM team; product design, support, introduction, and launch into volume production

Birger Engineering, Inc.
www.birger.com
Development of algorithms, electronics, and systems for digital imaging

BitSim AB
www.bitsim.com
Design services; DSP, video, 3G, and WLAN designs; high-speed prototyping boards

Bottom Line Technologies Inc.
www.bltinc.com
FPGA, product, and system design services; networking; data communications; telecommunications; video; PCI; signal processing; military; aerospace

CES Design Services @ Siemens Program and System Engineering
www.ces-designservices.com
Platform-oriented SoC/FPGA/PCB and firmware design in telecom, industrial, and video processing; customized solutions; concept; verification; prototypes

CG-CoreEl Programmable Solutions P. Ltd.
www.cg-coreel.com
Design services; telecom, networking, and processor IP and designs; turnkey FPGA; RTL design/validation

Chess iT/Embedded
www.chess.nl/
Embedded hardware and software design of products and services at the chip, board, and system level

Colorado Electronic Product Design, Inc.
www.cepdinc.com
Designing embedded systems with FPGAs for DSP; interfaces for medical, consumer, aerospace, and military

Comit Systems, Inc.
www.comit.com
High-end FPGA/SoC design services, including the integration of customer-developed and third-party IP

ControlNet India Pvt Ltd.
www.controlnetindia.com
ASIC/SoC Analog/RF FPGA design services; IP development; domains: Ethernet, IEEE1394, USB, security, PCI, WLAN, Infiniband

Convergent Design LLC
www.convergent-design.com
Audio/video design for professional/consumer products

DATA Respons ASA
www.datarespons.com
Professional design and development services for embedded hardware/software solutions; FPGA/digital hardware application specialists

Deltatec International Inc.
www.deltatec.be/
Design services; digital imaging applications; broadcast video, image processing, and image synthesis

Digicom Answers Pty. Ltd.
www.fpga.com.au/
Design services; experienced FPGA resources; design-on-demand; pre-design advice; reviews and problem solving

Digital Design Corporation (DDC)
www.digidescorp.com
Design services and products; image processing; video; communications; DSP; audio; automotive; controls; interfaces

Dillon Engineering, Inc.
www.dilloneng.com
Design services; FPGA-based DSP algorithms; high-bandwidth, real-time digital signal and image processing applications

DRS Tactical Systems West Inc. (formally Catalina Research)
www.catalinaresearch.com
Design services; FPGAs for DSP, including the manufacturer of Virtex reconfigurable computing boards

Eden Networks, Inc.
www.edennetworks.com
Turnkey design services; telecom and networking designs (Ethernet, SONET, ATM, VoIP)

Edgewood Technologies, LLC
www.edgewoodtechnologies.com
Design services; FPGAs for telecom, DSP, embedded processors

Enea Data AB
www.enea.se/
Telecommunication, data communication, DSP, RTOS, automotive, defense, high-reliability, and consumer electronics design

Enterpoint Ltd.
www.enterpoint.co.uk/
Design services; development boards; video, telecommunications, military, and high-speed design

ESD Inc.
www.esdnet.com
Design services; FPGAs for embedded systems, software/hardware design, PCB layout

Evatronix S.A.
www.evatronix.pl/
Design services; FPGA design (Virtex, Virtex-E, Spartan-II); embedded systems; SoC and SoPC designs

Evermore Systems Inc.
www.evermoresystems.com
Design services; FPGA designs for communications, audio/video, wireless, and home networking

ExaLinx
www.exalinx.com
Design services; ASIC emulation for high-speed communications and wireless applications; reference designs

Fidus Systems Inc.
www.fidus.ca/
Design services; FPGAs; communications; video; DSP; interface conversions; ASIC verification; hardware; PCB layout; software; SI/EMC

First Principles Technology, Inc.
www.fptek.com
Design/assembly/test services; FPGA + PCB + DSP/processor + system; imaging; video; test; satellite telecom; control

Flexibilis Oy
www.flexibilis.com
IP development; FPGA design services; Linux device drivers

Flextronics Design
www.flextronics.com/design/
Complete product design and development services for high-speed communications; reconfigurable computing; imaging; audio/video; military

GDA Technologies, Inc.
www.gdatech.com
A leading services company focused on designing systems, SoC, ASIC, FPGA, and IP

HCL Technologies
www.hcltech.com
High-speed board, ASIC, and FPGA design and verification services in avionics, medical, networking/telecom, and consumer electronics

Helion Technology Limited
www.heliontech.com
Design services and IP for FPGA with a focus on data security, wireless, and DSP

IDERS Inc.
www.iders.ca/
Contract electronic engineering and EMS resource, providing complete product cycles ranging from specifications to production

Image Technology Laboratory Corporation
www.gazogiken.co.jp/
Design services; FPGAs for image processing; design tools: EDK, System Generator, and Forge

Innovative Computer Technology
ictconn.com
Digital, analog, Xilinx FPGA, and software design solutions for customers in a wide variety of industries throughout the United States

iWave Systems Technologies Private Ltd.
www.iwavesystems.com
Embedded hardware and software design; board design; FPGA and ASIC development services

Liewenthal Electronics Ltd.
www.liewenthal.ee/
Embedded systems design services; development of software and hardware including Virtex and Spartan FPGAs

Logic Product Development
www.logicpd.com
Full-service product development, including industrial design, mechanical, electrical, and software engineering

M&M Consulting
hellman@mmcons.com
Design services; imaging; video and audio designs; high-density ASIC emulation

Memec Design
www.memecdesign.com/xilinx
Focused on FPGA design for Xilinx technology, we have completed more than 1,300 designs worldwide

Memondo Graphics
www.memondo.com
Implementation of digital signal processing, communications, and image processing solutions and algorithms over Xilinx FPGAs

Mikrokrets AS
www.mikrokrets.no/
FPGA development comprising telecommunication, network communications, and embedded systems; PCB development

Millogic Ltd.
www.millogic.com
Design services and synthesizable cores; expertise in PCI, video, imaging, DSP, compression, ASIC verification, and communications

Milstar Inc.
www.Milstar.co.il/
Design services and manufacturers (industrial and military specs); DSP; communication; video; audio; high-speed boards

Misarc srl - Agrate
www.misarc.com
Design services and manufacturers; telecommunications, SoC development, and test equipment boards using FPGA solutions

Multi Video Designs
www.mvd-fpga.com
Design services; DSP for video, telecommuications; SoC with Virtex-II/Pro and Spartan (PowerPC/MicroBlaze™); Xilinx training

Multiple Access Communications Ltd.
www.macltd.com
Design services; products and consultancy; DSP for wireless communications

Nallatech
www.nallatech.com
Embedded reconfigurable computers for high-performance applications in DSP, imaging, and defense; Xilinx Diamond XPERTS

NetModule AG
www.netmodule.ch/
Design services for system design; FPGA applications in telecommunication systems and medical electronics

NetQuest Corporation
www.netquestcorp.com
Turnkey engineering design and production services in advanced high-speed embedded communications

North Pole Engineering Inc.
www.northpoleengineering.com
Hardware and software design services for FPGA, ASIC, and embedded systems

Northwest Logic
www.nwlogic.com
PCI, PCI-X, and SDRAM controller IP; high-end FPGA design services

NovTech Engineering
www.novtech.com
Board-level and IP cores; turnkey design services; imaging; communication; PCI cores; encryption

NUVATION
www.nuvation.com
Design services; imaging and communications for defense/security, medical, consumer, and datacom/telecom markets

Plextek Ltd.
www.plextek.com
Design services; FPGA, DSP, radio, microwave, and microprocessor technologies for defense and communications

POLAR-Design
www.polar-design.de/
High-end FPGA design for telecom, networking, and DSP applications

Polybus Systems Corporation
www.polybus.com
Test patterns; customizable in-system FPGA test patterns; design services; networking and communications

Presco, Inc.
www.prescoinc.com
Twenty-five years of experience in custom electronics design/manufacturing for high-performance image processing, data communications, inkjet

Productivity Engineering GmbH
www.pe-gmbh.com
Design services; PCI design; microprocessor cores; obsolete component replacements; ASIC prototyping

Programall Technologies Inc.
www.ProgramallTechnologies.com
Design services; embedded systems targeting Virtex, Spartan, Virtex-II, and Virtex-II Pro FPGAs

R&D Consulting
972-9-7673074
FPGA development; video; imaging; graphics; DSP; communications

Rapid Prototypes, Inc.
www.FPGA.com
Design services; high-performance reconfigurable FPGA applications requiring maximum speed or density using physical design principles

RDLABS
www.rdlabs.com
FPGA/ASIC design; wireless communications; 802.11b/a IP cores; Matlab, Sysgen, ModelSim; instruments HP1661A, HP8594E, HP54520A

RightHand Technologies, Inc.
www.righthandtech.com
Design services; Virtex-II Pro FPGAs, boards, and software for video gaming, storage, medical, and wireless

Rising Edge Technology Inc.
www.risingedgetech.com
Complete design solutions from the ground up; FPGA interface designs

Riverside Machines Ltd.
www.riverside-machines.com
Design services and project management; DSP, wireless, network, and processor development; Verilog, VHDL, SystemC, Specman

Roke Manor Research
www.roke.co.uk/
Design services; IP, ATM, DSP, imaging, radar, and control solutions for FPGAs

Roman-Jones, Inc.
www.roman-jones.com
Design services; FPGAs for embedded microprocessors; PCI; DSP; low-cost 8051 softcore

Sapphire Computers, Inc.
drudolf@voyager.net
Engineering design consulting; high-speed digital and FPGA design

Siemens AG, Electronic Design House
www.eda-services.com
Design services; system-on-chip; ASICs; FPGAs; IPs; DSPs; boards; EMC; embedded software; prototypes

Silicon & Software Systems Ltd. (S3)
www.s3group.com
World-class electronics design company uniquely combining IC, FPGA, software, and hardware design expertise

Silicon Interfaces America Inc.
www.siliconinterfaces.com
Design services; develops IP in areas of networking, wireless, optical, datacom, interconnect, and microcontrollers

Silicon System Solutions P/L
www.silicon-systems.com
Design services; FPGAs for very high-speed applications including communications and DSP applications

Siscad S.p.A.
www.siscad.it/
Xilinx FPGA design services; IP development and integration

Smart Logic, Inc.
www.smart-logic.com
Design services; FPGAs for rapid prototyping including imaging, compression, and real-time control

SoleNet, Inc.
www.solenet.net
FPGA, board, and system designs for wireless, telecommunications, consumer, and reference design applications

so-logic electronic consulting
www.so-logic.net
Embedded systems (PowerPC, MicroBlaze); system-level design (Forge, System Generator, Handel-C); Xilinx training centers (Austria, Eastern Europe)

Styrex AB
www.styrex.se/
Design services for embedded systems; programmable logic, hardware development, and microprocessor programming

Synopsys Professional Services
www.synopsys.com
Design services to solve your design challenges in system-level design, RTL design, and physical design

Syntera AB
www.syntera.se/
Design services; FPGAs for radar, digital communications, telecom automotive, aerospace, and DSP

Tao of Digital, Inc.
www.taoofdigital.com
High-speed, large-gate-count SoC designs using IP cores for maximum modularity and efficiency

Technolution B.V.
www.technolution.nl/
Innovative hardware and software solutions

Tezzaron Semiconductor
www.tezzaron.com
Design services for portable computing, networking, and embedded systems

Thales Airborne Systems
www.thalesgroup.com/airbornesystems
Provider and integrator with high expertise in FPGA design for radars, DSP, and communications

TietoEnator R&D Services AB
www.TietoEnator.com
European design services for telecom, industrial IT, and automotive; Xilinx exclusive training provider in Scandinavia

V-Integration
www.V-Integration.com
Providing first-pass success in complex FPGA designs; applications include imaging, communications, and interface design

Williams Consulting, Inc.
chipw@wciatl.com
Embedded systems hardware and software; video; MPEG; control

AllianceEmbedded Members
www.xilinx.com/allianceembedded

See these AllianceEmbedded members for information about the following products (contact Xilinx for Wind River Xilinx Edition products):

Accelerated Technology® Division of Mentor Graphics
www.acceleratedtechnology.com
Nucleus® RTOS support for Xilinx MicroBlaze 32-bit soft processor

Avnet Design Services single board computers
www.em.avnet.com/home/
ADS Xilinx Virtex-II Pro Evaluation Kit

Corelis Inc.
www.corelis.com
CodeRunner JTAG debugger support for Virtex-II Pro embedded PowerPC

Express Logic
www.expresslogic.com
ThreadX® RTOS support for Xilinx MicroBlaze 32-bit soft processor

Green Hills Software
www.ghs.com
MULTI® debugger support for Virtex-II Pro embedded PowerPC; Green Hills™ Optimizing C Compiler support for Virtex-II Pro embedded PowerPC

MEMEC Design single board computers
www.insight-electronics.com/virtex2pro/
Virtex Series system boards

Micrium Technologies Corp.
www.ucos-ii.com
µC/OS-II RTOS support for Xilinx MicroBlaze 32-bit soft processor

Mind NV
www.mind.be/v2p/ecos
Mind NV eCos RTOS support for Virtex-II Pro embedded PowerPC

MontaVista Software
www.mvista.com
MontaVista® Linux® Professional Edition subscription support for Virtex-II Pro embedded PowerPC

Nohau Corporation
www.nohau.com
EMUL-MicroBlaze-PC debugger support for Xilinx MicroBlaze 32-bit soft processor

QNX Software
www.qnx.com
QNX® Neutrino® RTOS support for Virtex-II Pro embedded PowerPC

Wind River Systems
www.windriver.com
VxWorks® RTOS support for Virtex-II Pro embedded PowerPC; Diab™ Xilinx Edition (XE) C/C++ Compiler; SingleStep™ Xilinx Edition debugger; visionPROBE II Xilinx Edition; visionICE II Emulator; visionTRACE Emulator

XILINX®
The Programmable Logic Company℠

Xilinx Virtex™-II Series FPGAs

http://www.xilinx.com/devices/

Product Selection Matrix

				CLB Resources				Memory Resources			DSP	Clock Resources		I/O Features				Speed						
		Virtex-II Series EasyPath Solutions (see note 4)	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 (kbits)	# 18 kbits Block RAM	Total Block RAM (kbits)	# 18x18 Dedicated Multipliers	DCM Frequency (min/max)	# DCM Blocks (see note 3)	Digitally Controlled Impedance	Maximum Differential I/O Pairs	Maximum I/O	I/O Standards	Commercial Speed Grades (slowest to fastest)	Industrial Speed Grades (slowest to fastest)	Configuration Memory (Bits)	RocketIO™ Transceiver Blocks (3.125 Gbps)	RocketIO™ X Transceiver Blocks (10.3125 Gbps)	PowerPC™ Processor Blocks	EasyPath
Platform FPGAs	**Virtex-II Pro Family – 1.5 Volt**																							
VIRTEX-II PRO	XC2VP2		*	16 x 22	1,408	3,168	2816	44	12	216	12	24/420	4	YES	100	204	LDT-25, LVDS-25, LVDSEXT-25, BLVDS-25, ULVDS-25, LVPECL-25, LVCMOS25, LVCMOS18, LVCMOS15, PCI33, LVTTL, LVCMOS33, PCI-X, PCI66, GTL, GTL+, HSTL I (1.5V,1.8V), HSTL II (1.5V,1.8V), HSTL III (1.5V,1.8V), HSTL IV (1.5V,1.8V), SSTL2I, SSTL2II, SSTL18 I, SSTL18 II	-5 -6 -7	-5 -6	1.3M	4		0	
	XC2VP4		*	40 x 22	3,008	6,768	6,016	94	28	504	28	24/420	4	YES	172	348		-5 -6 -7	-5 -6	3.0M	4		1	
	XC2VP7		*	40 x 34	4,928	11,088	9,856	154	44	792	44	24/420	4	YES	196	396		-5 -6 -7	-5 -6	4.4M	8		1	
	XC2VP20		*	56 x 46	9,280	20,880	18,560	290	88	1,584	88	24/420	8	YES	276	564		-5 -6 -7	-5 -6	8.2M	8		2	
	XC2VP30	XCE2VP30	*	80 x 46	13,696	30,816	27,392	428	136	2448	136	24/420	8	YES	372	644		-5 -6 -7	-5 -6	11.3M	8		2	
	XC2VP40	XCE2VP40	*	88 x 58	19,392	43,632	38,784	606	192	3,456	192	24/420	8	YES	396	804		-5 -6 -7	-5 -6	15.5M	0 or 12		2	
	XC2VP50	XCE2VP50	*	88 x 70	23,616	53,136	47,232	738	232	4,176	232	24/420	8	YES	420	852		-5 -6 -7	-5 -6	19.0M	0 or 16		2	
	XC2VP70	XCE2VP70	*	104 x 82	33,088	74,448	66,176	1,034	328	5,904	328	24/420	8	YES	492	996		-5 -6 -7	-5 -6	25.6M	16 or 20		2	
	XC2VP100	XCE2VP100	*	120 x 94	44,096	99,216	88,192	1,378	444	7,992	444	24/420	12	YES	572	1,164		-5 -6 -7	-5 -6	33.5M	0 or 20		2	
	XC2VP125	XCE2VP125	*	136 x 106	55,616	125,136	111,232	1,738	556	10,008	556	24/420	12	YES	644	1,200		-5 -6 -7	-5 -6	42.7M	0, 20 or 24		4	
	Virtex-II Pro X Family – 1.5 Volt																							
VIRTEX-II PRO X	XC2VPX20		*	56 x 46	9,792	22,032	19,584	306	88	1584	88	24/420	8	YES	276	556	Same as above	-5 -6 -7	TBD	8.05M	0	8	1	
	XC2VPX70	XCE2VPX70	*	104 x 82	33,088	74,448	66,176	1034	308	5544	308	24/420	8	YES	492	996	Same as above	-5 -6 -7	TBD	25.60M	0	20	2	
	Virtex-II Family – 1.5 Volt																							
VIRTEX-II	XC2V40		40K	8 x 8	256	576	512	8	4	72	4	24/420	4	YES	44	88	LDT-25, LVPECL-33, LVDS-33, LVDS-25, LVDSEXT-33, LVDSEXT-25, BLVDS-25, ULVDS-25, LVTTL, LVCMOS33, LVCMOS25, LVCMOS18, LVCMOS15, PCI33, PCI66, PCI-X, GTL, GTL+, HSTL I, HSTL II, HSTL III, HSTL IV, SSTL2I, SSTL2II, SSTL3 I, SSTL3 II, AGP, AGP-2X	-4 -5 -6	-4 -5	0.4M	–	–	–	–
	XC2V80		80K	16 x 8	512	1,152	1,024	16	8	144	8	24/420	4	YES	60	120		-4 -5 -6	-4 -5	0.6M	–	–	–	–
	XC2V250		250K	24 x 16	1,536	3,456	3,072	48	24	432	24	24/420	8	YES	100	200		-4 -5 -6	-4 -5	1.7M	–	–	–	–
	XC2V500		500K	32 x 24	3,072	6,912	6,144	96	32	576	32	24/420	8	YES	132	264		-4 -5 -6	-4 -5	2.8M	–	–	–	–
	XC2V1000		1M	40 x 32	5,120	11,520	10,240	160	40	720	40	24/420	8	YES	216	432		-4 -5 -6	-4 -5	4.1M	–	–	–	–
	XC2V1500		1.5M	48 x 40	7,680	17,280	15,360	240	48	864	48	24/420	8	YES	264	528		-4 -5 -6	-4 -5	5.7M	–	–	–	–
	XC2V2000		2M	56 x 48	10,752	24,192	21,504	336	56	1,008	56	24/420	8	YES	312	624		-4 -5 -6	-4 -5	7.5M	–	–	–	–
	XC2V3000	XCE2V3000	3M	64 x 56	14,336	32,256	28,672	448	96	1,728	96	24/420	12	YES	360	720		-4 -5 -6	-4 -5	10.5M	–	–	–	–
	XC2V4000	XCE2V4000	4M	80 x 72	23,040	51,840	46,080	720	120	2,160	120	24/420	12	YES	456	912		-4 -5 -6	-4 -5	15.7M	–	–	–	–
	XC2V6000	XCE2V6000	6M	96 x 88	33,792	76,032	67,584	1,056	144	2,592	144	24/420	12	YES	552	1,104		-4 -5 -6	-4 -5	21.9M	–	–	–	–
	XC2V8000	XCE2V8000	8M	112 x 104	46,592	104,832	93,184	1,456	168	3,024	168	24/420	12	YES	554	1,108		-4 -5	-4 -5	29.1M	–	–	–	–

* Logic cell counts are a more meaningful measurement of density for the Virtex-II Pro family since system gate count does not take into consideration the benefits of the immersed special blocks such as PowerPC processors and multi-gigabit transceivers.

Notes: 1. System Gates include 20-30% of CLBs used as RAM. 2. Logic cell = One 4-Input Look Up Table (LUT) + Flip Flop + Carry Logic. 3. DCM = Digital Clock Management. 4. Virtex-II Series EasyPath solution available to provide a no risk, no effort cost reduction path for volume production.

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/O[1,2]

		Virtex-II Pro (1.5V)									
		XC2VP2	XC2VP4	XC2VP7	XC2VP20	XC2VP30	XC2VP40	XC2VP50	XC2VP70	XC2VP100	XC2VP125
Pins[3]	Body Size	204	348	396	564	692	804	852	996	1164	1200
Chip Scale Packages (CS) – wire-bond chip-scale BGA (0.8 mm ball spacing)											
144	12 x 12 mm										
BGA Packages (BG) – wire-bond standard BGA (1.27 mm ball spacing)											
575	31 x 31 mm										
728	35 x 35 mm										
FGA Packages (FG) – wire-bond fine-pitch BGA (1.0 mm ball spacing)											
256	17 x 17 mm	140	140								
456[4]	23 x 23 mm	156	248	248							
676[4]	27 x 27 mm				404	416	416				
FFA Packages (FF) – flip-chip fine-pitch BGA (1.0 mm ball spacing)											
672	27 x 27 mm	204	348	396							
896[5]	31 x 31 mm			396	556	556					
1152[5]	35 x 35 mm				564	564	692	692			
1148[6]	35 x 35 mm						804	812			
1517	40 x 40 mm							852	964		
1704	42.5 x 42.5 mm								996	1040	1040
1696[6]	42.5 x 42.5 mm									1164	1200
BFA Packages (BF) – flip-chip fine-pitch BGA (1.27 mm ball spacing)											
957	40 x 40 mm										

		Virtex-II Pro X (1.5V)	
		XC2VPX20	XC2VPX70
Pins[3]	Body Size	556	996
Chip Scale Packages (CS) – wire-bond chip-scale BGA (0.8 mm ball spacing)			
144	12 x 12 mm		
BGA Packages (BG) – wire-bond standard BGA (1.27 mm ball spacing)			
575	31 x 31 mm		
728	35 x 35 mm		
FGA Packages (FG) – wire-bond fine-pitch BGA (1.0 mm ball spacing)			
256	17 x 17 mm		
456[4]	23 x 23 mm		
676[4]	27 x 27 mm		
FFA Packages (FF) – flip-chip fine-pitch BGA (1.0 mm ball spacing)			
672	27 x 27 mm		
896[5]	31 x 31 mm	556	
1152[5]	35 x 35 mm		
1148[6]	35 x 35 mm		
1517	40 x 40 mm		
1704	42.5 x 42.5 mm		996
1696[6]	42.5 x 42.5 mm		
BFA Packages (BF) – flip-chip fine-pitch BGA (1.27 mm ball spacing)			
957	40 x 40 mm		

		Virtex-II (1.5V)										
		XC2V40	XC2V80	XC2V250	XC2V500	XC2V1000	XC2V1500	XC2V2000	XC2V3000	XC2V4000	XC2V6000	XC2V8000
Pins[3]	Body Size	88	120	200	264	432	528	624	720	912	1104	1296
Chip Scale Packages (CS) – wire-bond chip-scale BGA (0.8 mm ball spacing)												
144	12 x 12 mm	88	92	92								
BGA Packages (BG) – wire-bond standard BGA (1.27 mm ball spacing)												
575	31 x 31 mm					328	392	408				
728	35 x 35 mm								516			
FGA Packages (FG) – wire-bond fine-pitch BGA (1.0 mm ball spacing)												
256	17 x 17 mm	88	120	172	172	172						
456[4]	23 x 23 mm			200	264	324						
676[4]	27 x 27 mm						392	456	484			
FFA Packages (FF) – flip-chip fine-pitch BGA (1.0 mm ball spacing)												
672	27 x 27 mm											
896[5]	31 x 31 mm					432	528	624				
1152[5]	35 x 35 mm								720	824	824	824
1148[6]	35 x 35 mm											
1517	40 x 40 mm									912	1104	1108
1704	42.5 x 42.5 mm											
1696[6]	42.5 x 42.5 mm											
BFA Packages (BF) – flip-chip fine-pitch BGA (1.27 mm ball spacing)												
957	40 x 40 mm							624	684	684	684	

	Transceiver Blocks											
	RocketIO (3.125 Gbps)										RocketIO X (10 Gbps)	
Package	XC2VP2	XC2VP4	XC2VP7	XC2VP20	XC2VP30	XC2VP40	XC2VP50	XC2VP70	XC2VP100	XC2VP125	XC2VPX20	XC2VPX70
FG256	4	4										
FG456	4	4	8									
FG676				8	8	8						
FG672	4	4	8									
FF896			8	8	8						8	
FF1152				8	8	12	16					
FF1148[6]						0	0					
FF1517							16	16				
FF1704								20	20	24		20
FF1696[6]									0	0		

Notes: 1. Numbers in table indicate maximum number of user I/Os.
2. The number of I/Os for RocketIO MGTs are not included in this table.
3. Within the same family, all devices in a particular package are pin-out (footprint) compatible.
4. Virtex-II packages FG456 and FG676 are also footprint compatible.
5. Virtex-II packages FF896 and FF1152 are also footprint compatible.
6. RocketIO unavailable in this package.

Important: Verify all data in this document with the device data sheets found at http://www.xilinx.com/partinfo/databook.htm

XILINX®
The Programmable Logic Company℠

Xilinx Spartan™ Series FPGAs

http://www.xilinx.com/devices/

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	I/O Standards	Commercial Speed Grades (slowest to fastest)	Industrial Speed Grades (slowest to fastest)	Configuration Memory (Bits)
Spartan-3 Family – 1.2 Volt (see note 3)																				
XC3S50	50K	16 x 12	768	1,728	1,536	12K	4	72K	4	25/326	2	YES	YES	YES	56	124	Single-ended LVTTL, LVCMOS3.3/2.5/1.8/ 1.5/1.2, PCI 3.3V – 32/64-bit 33MHz, SSTL2 Class I & II, SSTL18 Class I, HSTL Class I, III, HSTL1.8 Class I, II & III, GTL, GTL+ Differential LVDS2.5, Bus LVDS2.5, Ultra LVDS2.5, LVDS_ext2.5, RSDS, LDT2.5, LVPECL	-4 -5	-4	.4M
XC3S200	200K	24 x 20	1,920	4,320	3,840	30K	12	216K	12	25/326	4	YES	YES	YES	76	173		-4 -5	-4	1.0M
XC3S400	400K	32 x 28	3,584	8,064	7,168	56K	16	288K	16	25/326	4	YES	YES	YES	116	264		-4 -5	-4	1.7M
XC3S1000	1000K	48 x 40	7,680	17,280	15,360	120K	24	432K	24	25/326	4	YES	YES	YES	175	391		-4 -5	-4	3.2M
XC3S1500	1500K	64 x 52	13,312	29,952	26,624	208K	32	576K	32	25/326	4	YES	YES	YES	221	487		-4 -5	-4	5.2M
XC3S2000	2000K	80 x 64	20,480	46,080	40,960	320K	40	720K	40	25/326	4	YES	YES	YES	270	565		-4 -5	-4	7.7M
XC3S4000	4000K	96 x 72	27,648	62,208	55,296	432K	96	1,728K	96	25/326	4	YES	YES	YES	312	712		-4 -5	-4	11.3M
XC3S5000	5000K	104 x 80	33,280	74,880	66,560	520K	104	1,872K	104	25/326	4	YES	YES	YES	344	784		-4 -5	-4	13.3M

Notes: 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

XILINX®
The Programmable Logic Company℠

Xilinx Spartan™ Series FPGAs

http://www.xilinx.com/devices/

Product Selection Matrix

		CLB Resources				Memory Resources			CLK Resources				I/O Features			Speed				
		System Gates (see note 1)	CLB Array (Row x Col)	Number of Slices	Logic Cells (see notes 2 and 3)	CLB Flip-Flops	Max. Distributed RAM Bits	# Block RAM	Block RAM (bits)	DLL Frequency (min/max)	# DLLs	Frequency Synthesis	Phase Shift	Number of Differential I/O Pairs	Maximum I/O	I/O Standards	Commercial Speed Grades (slowest to fastest)	Industrial Speed Grade (slowest to fastest)	Automotive Q-Grade Speed Grade	Configuration Memory (Bits)	Automotive Q-Grade Solutions (see note 4)
	Spartan-IIE Family – 1.8 Volt																				
SPARTAN-IIE	XC2S50E	50K	16 x 24	768	1,728	1,536	24K	8	32K	25/320	4	YES	YES	83	182	LVTTL, LVCMOS2, LVCMOS18, PCI33, PCI66, GTL, GTL+, HSTL I, HSTL III, HSTL IV, SSTL3 I, SSTL3 II, SSTL2 I, SSTL2 II, AGP-2X, CTT, LVDS, BLVDS, LVPECL	-6 -7	-6	-6	0.6M	✔
	XC2S100E	100K	20 x 30	1,200	2,700	2,400	37K	10	40K	25/320	4	YES	YES	86	202		-6 -7	-6	-6	0.9M	✔
	XC2S150E	150K	24 x 36	1,728	3,888	3,456	54K	12	48K	25/320	4	YES	YES	114	265		-6 -7	-6	-6	1.1M	✔
	XC2S200E	200K	28 x 42	2,352	5,292	4,704	73K	14	56K	25/320	4	YES	YES	120	289		-6 -7	-6	-6	1.4M	✔
	XC2S300E	300K	32 x 48	3,072	6,912	6,144	96K	16	64K	25/320	4	YES	YES	120	329		-6 -7	-6	-6	1.9M	✔
	XC2S400E	400K	40 x 60	4,800	10,800	9,600	150K	40	160K	25/320	4	YES	YES	172	410		-6 -7	-6	-6	2.7M	✔
	XC2S600E	600K	48 x 72	6,912	15,552	13,824	216K	72	288K	25/320	4	YES	YES	205	514		-6 -7	-6	-6	4.0M	✔
	Spartan-II Family – 2.5 Volt																				
SPARTAN-II	XC2S15	15K	8 x 12	192	432	384	6K	4	16K	25/200	4	YES	YES	NA	86	LVTTL, LVCMOS2, PCI33 (3.3V & 5V), PCI66 (3.3V), GTL, GTL+, HSTL I, HSTL III, HSTL IV, SSTL3 I, SSTL3 II, SSTL2 I, SSTL2 II, AGP-2X, CTT	-5 -6	-5	-5	0.2M	✔
	XC2S30	30K	12 x 18	432	972	864	13.5K	6	24K	25/200	4	YES	YES	NA	132		-5 -6	-5	-5	0.4M	✔
	XC2S50	50K	16 x 24	768	1,728	1,536	24K	8	32K	25/200	4	YES	YES	NA	176		-5 -6	-5	-5	0.6M	✔
	XC2S100	100K	20 x 30	1,200	2,700	2,400	37.5K	10	40K	25/200	4	YES	YES	NA	196		-5 -6	-5	-5	0.8M	✔
	XC2S150	150K	24 x 36	1,728	3,888	3,456	54K	12	48K	25/200	4	YES	YES	NA	260		-5 -6	-5	-5	1.1M	✔
	XC2S200	200K	28 x 42	2,352	5,292	4,704	73.5K	14	56K	25/200	4	YES	YES	NA	284		-5 -6	-5	-5	1.4M	✔
	Spartan-XL Family – 3.3 Volt																				
SPARTAN-XL	XCS05XL	5K	10 x 10	100	238	200	3.1K	NA	NA	NA	NA	NA	NA	NA	77		-4 -5	-4	-4	0.05M	✔
	XCS10XL	10K	14 x 14	196	466	392	6.1K	NA	NA	NA	NA	NA	NA	NA	112		-4 -5	-4	-4	0.09M	✔
	XCS20XL	20K	20 x 20	400	950	800	12.5K	NA	NA	NA	NA	NA	NA	NA	160		-4 -5	-4	-4	0.18M	✔
	XCS30XL	30K	24 x 24	576	1,368	1,152	18.0K	NA	NA	NA	NA	NA	NA	NA	192		-4 -5	-4	-4	0.25M	✔
	XCS40XL	40K	28 x 28	784	1,862	1,568	24.5K	NA	NA	NA	NA	NA	NA	NA	224		-4 -5	-4	-4	0.33M	✔

Notes: 1. System Gates include 20-30% of CLBs used as RAM.
2. Logic cell = (1) 4-Input (LUT) and a register.
3. For Spartan-IIE/II/XL, a Logic Cell is defined as a 4-input LUT + a register.
4. Automotive Q-Grade Solutions are qualified to -40°C to +125°C junction temperature for FPGAs. Q-Grade products 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

XILINX®
The Programmable Logic Company℠

Xilinx Spartan™ Series FPGAs

http://www.xilinx.com/devices/

Package Options and User I/O[1]

		Spartan-3 (1.2V)								
			XC3S50	XC3S200	XC3S400	XC3S1000	XC3S1500	XC3S2000	XC3S4000	XC3S5000
Pins	Body Size	I/Os	124	173	264	391	487	565	712	784
PLCC Packages (PC) – wire-bond plastic chip carrier (1.27 mm lead spacing)										
84	29.3 x 29.3 mm									
PQFP Packages (PQ) – wire-bond plastic QFP (0.5 mm lead spacing)										
208	28 x 28 mm		124	141	141					
240	32 x 32 mm									
VQFP Packages (VQ) – very thin TQFP (0.5 mm lead spacing)										
100	14 x 14 mm		63	63						
TQFP Packages (TQ) – thin QFP (0.5 mm lead spacing)										
144	20 x 20 mm		97	97	97					
Chip Scale Packages (CS) – wire-bond chip-scale BGA (0.8 mm ball spacing)										
144	12 x 12 mm									
280	16 x 16 mm									
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)										
256	17 x 17 mm									
456	23 x 23 mm				264	333	333			
676	27 x 27 mm					391	487	489		
900	31 x 31 mm							565	633	633
1156	35 x 35 mm								712	784
BGA Packages (BG) – wire-bond standard BGA (1.27 mm ball spacing)										
256	27 x 27 mm									

	Spartan-IIE (1.8V)							
	XC2S50E	XC2S100E	XC2S150E	XC2S200E	XC2S300E	XC2S400E	XC2S600E
I/Os	182	202	265	289	329	410	514
PLCC Packages (PC) – wire-bond plastic chip carrier (1.27 mm lead spacing)							
84							
PQFP Packages (PQ) – wire-bond plastic QFP (0.5 mm lead spacing)							
208	146	146	146	146	146		
240							
VQFP Packages (VQ) – very thin TQFP (0.5 mm lead spacing)							
100							
TQFP Packages (TQ) – thin QFP (0.5 mm lead spacing)							
144	102	102					
Chip Scale Packages (CS) – wire-bond chip-scale BGA (0.8 mm ball spacing)							
144							
280							
FGA Packages (FT) – wire-bond fine-pitch thin BGA (1.0 mm ball spacing)							
256	182	182	182	182	182	182	
FGA Packages (FG) – wire-bond fine-pitch BGA (1.0 mm ball spacing)							
256							
456		202	265	289	329	329	329
676						410	514
900							
1156							
BGA Packages (BG) – wire-bond standard BGA (1.27 mm ball spacing)							
256							

	Spartan-II (2.5V)					
	XC2S15	XC2S30	XC2S50	XC2S100	XC2S150	XC2S200
I/Os	86	132	176	196	260	284
PLCC Packages (PC) – wire-bond plastic chip carrier (1.27 mm lead spacing)						
84						
PQFP Packages (PQ) – wire-bond plastic QFP (0.5 mm lead spacing)						
208		132	140	140	140	140
240						
VQFP Packages (VQ) – very thin TQFP (0.5 mm lead spacing)						
100	60	60				
TQFP Packages (TQ) – thin QFP (0.5 mm lead spacing)						
144	86	92	92	92		
Chip Scale Packages (CS) – wire-bond chip-scale BGA (0.8 mm ball spacing)						
144	86	92				
280						
FGA Packages (FT) – wire-bond fine-pitch thin BGA (1.0 mm ball spacing)						
256						
FGA Packages (FG) – wire-bond fine-pitch BGA (1.0 mm ball spacing)						
256			176	176	176	176
456				196	260	284
676						
900						
1156						
BGA Packages (BG) – wire-bond standard BGA (1.27 mm ball spacing)						
256						

	Spartan-XL (3.3V)				
	XCS05XL	XCS10XL	XCS20XL	XCS30XL	XCS40XL
I/Os	77	112	160	192	224
PLCC Packages (PC) – wire-bond plastic chip carrier (1.27 mm lead spacing)					
84	61	61			
PQFP Packages (PQ) – wire-bond plastic QFP (0.5 mm lead spacing)					
208			160	169	169
240				192	192
VQFP Packages (VQ) – very thin TQFP (0.5 mm lead spacing)					
100	77	77	77	77	
TQFP Packages (TQ) – thin QFP (0.5 mm lead spacing)					
144		112	113	113	
Chip Scale Packages (CS) – wire-bond chip-scale BGA (0.8 mm ball spacing)					
144		112	113		
280				192	224
FGA Packages (FT) – wire-bond fine-pitch thin BGA (1.0 mm ball spacing)					
256					
FGA Packages (FG) – wire-bond fine-pitch BGA (1.0 mm ball spacing)					
256					
456					
676					
900					
1156					
BGA Packages (BG) – wire-bond standard BGA (1.27 mm ball spacing)					
256				192	205

Note 1: Numbers in table indicate maximum number of user I/Os

Automotive products are highlighted: -40C to +125C junction temperature for FPGAs.
Automotive Q-Grade products will be available in Spartan-3 family 2H2004.
For more information on IQ Solutions please visit **http://www.xilinx.com/automotive**

Xilinx IQ Solutions for Automotive Intelligence

Important: Verify all data in this document with the device data sheets found at http://www.xilinx.com/partinfo/databook.htm

XILINX®
The Programmable Logic Company℠

Xilinx Configuration Storage Solutions

http://www.xilinx.com/configsolns

Product Selection and Package Option Matrix

Platform Flash Device Cross Reference

Platform Flash	PROM Solution
Spartan-3	
XC3S50	XCF01S
XC3S200	XCF01S
XC3S400	XCF02S
XC3S1000	XCF04S
XC3S1500	XCF08P
XC3S2000	XCF08P
XC3S4000	XCF16P
XC3S5000	XCF16P
Spartan-IIE	
XC2S50E	XCF01S
XC2S100E	XCF01S
XC2S150E	XCF02S
XC2S200E	XCF02S
XC2S300E	XCF02S
XC2S400E	XCF04S
XC2S600E	XCF04S
Spartan-II	
XC2S15	XCF01S
XC2S30	XCF01S
XC2S50	XCF01S
XC2S100	XCF01S
XC2S150	XCF01S
XC2S200	XCF02S
Spartan-XL	
XCS05XL	XCF01S
XCS10XL	XCF01S
XCS20XL	XCF01S
XCS30XL	XCF01S
XCS40XL	XCF01S

Platform Flash	PROM Solution
Virtex-II Pro	
XC2VP2	XCF02S
XC2VP4	XCF04S
XC2VP7	XCF08P
XC2VP20	XCF08P
XC2VP30	XCF16P
XC2VP40	XCF16P
XC2VP50	XCF32P
XC2VP70	XCF32P
XC2VP100	XCF32P*
XC2VP125	XCF32P**
Virtex-II	
XC2V40	XCF01S
XC2V80	XCF01S
XC2V250	XCF02S
XC2V500	XCF04S
XC2V1000	XCF04S
XC2V1500	XCF08P
XC2V2000	XCF08P
XC2V3000	XCF16P
XC2V4000	XCF16P
XC2V6000	XCF32P
XC2V8000	XCF32P

* Assumes bitstream compression is used (XCF32P + XCF01S without compression).
** Assumes bitstream compression is used (XCF32P + XCF16P without compression).

Note: For information regarding legacy PROMs, visit **http://www.xilinx.com/legacyproms**

Platform Flash Family Features and Packages

	XCF01S	XCF02S	XCF04S	XCF08P	XCF16P	XCF32P
Density	1 Mb	2 Mb	4 Mb	8 Mb	16 Mb	32 Mb
JTAG Prog	Y	Y	Y	Y	Y	Y
Serial Configuration	Y	Y	Y	Y	Y	Y
SelectMap Configuration				Y	Y	Y
Compression				Y	Y	Y
Design Rev				Y	Y	Y
VCC (V)	3.3	3.3	3.3	1.8	1.8	1.8
VCCO (V)	1.8 – 3.3	1.8 – 3.3	1.8 – 3.3	1.5 – 3.3	1.5 – 3.3	1.5 – 3.3
VCCJ (V)	1.8 – 3.3	1.8 – 3.3	1.8 – 3.3	1.5 – 3.3	1.5 – 3.3	1.5 – 3.3
Clock (MHz)	33	33	33	40	40	40
Package	VO20	VO20	VO20	FS48	FS48	FS48
				VO48	VO48	VO48

For multiple FPGA Configuration and for designs utilizing system level features, use SystemACE™.

SystemACE	Memory Density	Number of Components	Min. Board Space	Compression	FPGA Config. Mode	Multiple Designs	Software Storage	Removable	IRL Hooks	Max. Config. Speed	Non-Volatile Media
SystemACE CF	Up to 8 Gbit	2	25 cm^2	No	JTAG	Unlimited	Yes	Yes	Yes	30 Mbit/sec	CompactFlash
SystemACE SC	16 Mbit 32 Mbit 64 Mbit	3	Custom	Yes	SelectMAP (up to 4 FPGA) Slave-Serial (up to 8 FPGA chains)	Up to 8	No	No	Yes	152 Mbit/sec	AMD Flash Memory

Important: Verify all data in this document with the device data sheets found at http://www.xilinx.com/partinfo/databook.htm

Product Selection Matrix – CoolRunner™ Series

		System Gates	Macrocells	Product Terms per Macrocell	Input Voltage Compatible	Output Voltage Compatible	I/O Features: Maximum I/O	I/O Features: I/O Banking	Speed: Min. Pin-to-Pin Logic Delay (ns)	Speed: Commercial Speed Grades (fastest to slowest)	Speed: Industrial Speed Grades (fastest to slowest)	Speed: IQ Speed Grade	Clocking: Global Clocks	Clocking: Product Term Clocks per Function Block
	CoolRunner-II Family – 1.8 Volt													
CoolRunner-II	XC2C32	750	32	40	1.5/1.8/2.5/3.3	1.5/1.8/2.5/3.3	33	1	3	-3 -4 -6	-6	-6	3	17
	XC2C64	1500	64	40	1.5/1.8/2.5/3.3	1.5/1.8/2.5/3.3	64	1	4	-4 -5 -7	-7	-7	3	17
	XC2C128	3000	128	40	1.5/1.8/2.5/3.3	1.5/1.8/2.5/3.3	100	2	4.5	-4 -6 -7	-7	-7	3	17
	XC2C256	6000	256	40	1.5/1.8/2.5/3.3	1.5/1.8/2.5/3.3	184	2	5	-5 -6 -7	-7	-7	3	17
	XC2C384	9000	384	40	1.5/1.8/2.5/3.3	1.5/1.8/2.5/3.3	240	4	6	-6 -7 -10	-10	-10	3	17
	XC2C512	12000	512	40	1.5/1.8/2.5/3.3	1.5/1.8/2.5/3.3	270	4	6	-6 -7 -10	-10	-10	3	17
	CoolRunner XPLA3 Family – 3.3 Volt													
CoolRunner XPLA3	XCR3032XL	750	32	48	3.3/5	3.3	36		5	-5 -7 -10	-7 -10	-10	4	16
	XCR3064XL	1500	64	48	3.3/5	3.3	68		6	-6 -7 -10	-7 -10	-10	4	16
	XCR3128XL	3000	128	48	3.3/5	3.3	108		6	-6 -7 -10	-7 -10	-10	4	16
	XCR3256XL	6000	256	48	3.3/5	3.3	164		7.5	-7 -10 -12	-10 -12	-12	4	16
	XCR3384XL	9000	384	48	3.3/5	3.3	220		7.5	-7 -10 -12	-10 -12	-12	4	16
	XCR3512XL	12000	512	48	3.3/5	3.3	260		7.5	-7 -10 -12	-10 -12	-12	4	16

Package Options and User I/O

Pins	Body Size	CoolRunner-II: XC2C32	XC2C64	XC2C128	XC2C256	XC2C384	XC2C512	CoolRunner XPLA3: XCR3032XL	XCR3064XL	XCR3128XL	XCR3256XL	XCR3384XL	XCR3512XL
PLCC Packages (PC) – wire-bond plastic chip carrier (1.27 mm lead spacing)													
44	16.5 x 16.5 mm	33	33					36	36				
PQFP Packages (PQ) – wire-bond plastic QFP (0.5 mm lead spacing)													
208	28 x 28 mm				173	173	173				164	172	180
VQFP Packages (VQ) – very thin TQFP (0.5 mm lead spacing)													
44	10 x 10 mm	33	33					36	36				
100	14 x 14 mm		64	80	80				68	84			
TQFP Packages (TQ) – thin QFP (0.5 mm lead spacing)													
144	20 x 20 mm			100	118	118				108	120	118*	
Chip Scale Packages (CP) – wire-bond chip-scale BGA (0.5 mm ball spacing)													
56	6 x 6 mm	33	45						48				
132	8 x 8 mm			100	106								
Chip Scale Packages (CS) – wire-bond chip-scale BGA (0.8 mm ball spacing)													
48	7 x 7 mm							36	40				
144	12 x 12 mm									108			
280	16 x 16 mm										164		
FGA Packages (FT) – wire-bond fine-pitch thin BGA (1.0 mm ball spacing)													
256	17 x 17 mm				184	212	212				164	212	212
FBGA Packages (FG) – wire-bond fine-line BGA (1.0 mm ball spacing)													
324	23 x 23 mm					240	270					220	260

* JTAG pins and port enable are not pin compatible in this package for this member of the family.

Automotive products are highlighted: -40C to +125C ambient temperature for CPLDs.

Xilinx IQ Solutions for Automotive Intelligence

XILINX®
The Programmable Logic Company℠

Xilinx CPLD

http://www.xilinx.com/devices

Product Selection Matrix – 9500 Series

	System Gates	Macrocells	Product Terms per Macrocell	Input Voltage Compatible	Output Voltage Compatible	I/O Features: Maximum I/O	I/O Features: I/O Banking	Speed: Min. Pin-to-Pin Logic Delay (ns)	Speed: Commercial Speed Grades (fastest to slowest)	Speed: Industrial Speed Grades (fastest to slowest)	Speed: IQ Speed Grade	Clocking: Global Clocks	Clocking: Product Term Clocks per Function Block
XC9500XV Family – 2.5 Volt													
XC9536XV	800	36	90	2.5/3.3	1.8/2.5/3.3	36	1	5	-5 -7	-7	NA	3	18
XC9572XV	1600	72	90	2.5/3.3	1.8/2.5/3.3	72	1	5	-5 -7	-7	NA	3	18
XC95144XV	3200	144	90	2.5/3.3	1.8/2.5/3.3	117	2	5	-5 -7	-7	NA	3	18
XC95288XV	6400	288	90	2.5/3.3	1.8/2.5/3.3	192	4	6	-6 -7 -10	-7 -10	NA	3	18
XC9500XL Family – 3.3 Volt													
XC9536XL	800	36	90	2.5/3.3/5	2.5/3.3	36		5	-5 -7 -10	-7 -10	-10	3	18
XC9572XL	1600	72	90	2.5/3.3/5	2.5/3.3	72		5	-5 -7 -10	-7 -10	-10	3	18
XC95144XL	3200	144	90	2.5/3.3/5	2.5/3.3	117		5	-5 -7 -10	-7 -10	NA	3	18
XC95288XL	6400	288	90	2.5/3.3/5	2.5/3.3	192		6	-6 -7 -10	-7 -10	NA	3	18
XC9500 Family – 5 Volt													
XC9536	800	36	90	5	5	36		10	-5 -6 -10 -15	-7 -10 -15	-15	3	18
XC9572	1600	72	90	5	5	72		10	-7 -10 -15	-10 -15	-15	3	18
XC95108	2400	108	90	5	5	108		10	-7 -10 -15 -20	-7 -10 -15 -20	NA	3	18
XC95144	3200	144	90	5	5	133		10	-7 -10 -15	-10 -15	NA	3	18
XC95216	4800	216	90	5	5	166		10	-10 -15 -20	-10 -15 -20	NA	3	18
XC95288	6400	288	90	5	5	192		10	-10 -15 -20	-15 -20	NA	3	18

Package Options and User I/O

		XC9500XV				XC9500XL				XC9500					
Pins	Body Size	XC9536XV	XC9572XV	XC95144XV	XC95288XV	XC9536XL	XC9572XL	XC95144XL	XC95288XL	XC9536	XC9572	XC95108	XC95144	XC95216	XC95288
PLCC Packages (PC) – wire-bond plastic chip carrier (1.27 mm lead spacing)															
44	16.5 x 16.5 mm	34	34			34	34			34	34				
84	29.3 x 29.3 mm										69	69			
PQFP Packages (PQ) – wire-bond plastic QFP (0.5 mm lead spacing)															
100	18.85 x 12.35 mm										72	81	81		
160	28.0 x 28.0 mm											108	133	133	168
208	28.0 x 28.0 mm				168				168					166	168
VQFP Packages (VQ) – very thin TQFP (0.5 mm lead spacing)															
44	10 x 10 mm	34	34			34	34			34					
64	10 x 10 mm					36	52								
TQFP Packages (TQ) – thin QFP (0.5 mm lead spacing)															
100	14 x 14 mm		72	81			72	81			72	81	81		
144	20 x 20 mm			117	117			117	117						
Chip Scale Packages (CP) – wire-bond chip-scale BGA (0.5 mm ball spacing)															
56	6 x 6 mm														
132	8 x 8 mm														
Chip Scale Packages (CS) – wire-bond chip-scale BGA (0.8 mm ball spacing)															
48	7 x 7 mm	36	38			36	38			34					
144	12 x 12 mm			117				117							
280	16 x 16 mm				192				192						
BGA Packages (BG) – wire-bond standard BGA (1.27 mm ball spacing)															
256	27 x 27 mm								192						
352	35.0 x 35.0 mm								192					166	192
FGA Packages (FT) – wire-bond fine-pitch thin BGA (1.0 mm ball spacing)															
256	17 x 17 mm														
FBGA Packages (FG) – wire-bond fine-line BGA (1.0 mm ball spacing)															
256	17 x 17 mm				192				192						
324	23 x 23 mm														

Automotive products are highlighted: -40C to +125C ambient temperature for CPLDs.

Xilinx IQ Solutions for Automotive Intelligence

XILINX®
The Programmable Logic Company℠

Xilinx Defense and Aerospace

http://www.xilinx.com/devices

QPRO

Device	SMD	Voltage	System Gates	Logic Cells	RAMbus	CFG Bits	MULTs	DLLs	Flip-Flops	Max I/Os	Packages
XC3020*	5962-89948	5	1500	256	–	14779	–	–	256	64	PG84, CB100
XC3030*	None	5	2000	360	–	22176	–	–	360	80	PG84
XC3042*	5962-89713	5	3000	480	–	30784	–	–	480	96	PG84, PG132, CB100
XC3064*	None	5	4500	688	–	46064	–	–	688	120	PG132
XC3090*	5962-89823	5	6000	928	–	64160	–	–	928	144	PG175, CB164
XC4005*	5962-92252	5	9K	466	6272	151960	–	–	616	112	PG156, CB164
XC4010*	5962-92305	5	20K	950	12800	283424	–	–	1120	160	PG191, CB196
XC4013*	5962-94730	5	30K	1368	18438	393632	–	–	1536	192	PG223, CB228
XQ4005E	5962-97522	5	9K	466	6272	95008	–	–	616	112	PG156, CB164
XQ4010E	5962-97523	5	20K	950	12800	178144	–	–	1120	160	HQ208, PG191, CB196
XQ4013E	5962-97524	5	30K	1368	18438	247968	–	–	1536	192	HQ240, PG223, CB228
XQ4025E	5962-97525	5	45K	2432	32768	422176	–	–	2560	256	PG299, CB228
XQ4028EX	5962-98509	5	18K-50K	2432	32768	668184	–	–	2560	256	HQ240, BG352, PG299, CB228
XQ4013XL	5962-98513	3.3	10-30K	1368	18K	393632	–	–	1536	192	PQ240, BG256, PG223, CB228
XQ4036XL	5962-98510	3.3	22-65K	3078	42K	832528	–	–	3168	288	HQ240, BG352, PG411, CB228
XQ4062XL	5962-98511	3.3	40-130K	5472	74K	1433864	–	–	5376	384	HQ240, BG432, PG475, CB228
XQ4085XL	None	3.3	55-180K	7448	100K	1924992	–	–	7185	448	HQ240, BG432, CB228
XQV100	None	2.5	108904	2700	40K	781248	–	4	2400	180	PQ240, BG256, CB228
XQV300	5962-99572	2.5	322970	6912	64K	1751840	–	4	6144	316	PQ240, BG352, BG432, CB228
XQV600	5962-99573	2.5	661111	15552	96K	3608000	–	4	13824	512	HQ240, BG432, CB228
XQV1000	5962-99574	2.5	1124022	27648	128K	6127776	–	4	24576	512	BG560, CG560
XQV600E	None	1.8	986K	15552	288K	3961632	–	8	13824	512	BG432, CB228
XQV1000E	None	1.8	1569K	27648	384K	6587520	–	8	24576	660	BG560, CG560
XQV2000E	None	1.8	2542K	43200	640K	10159648	–	8	38400	804	BG560, FG1156
XQ2V1000	TBD	1.5	1000K	11520	720	4082592	40	8	10240	432	FG456, BG575
XQ2V3000	TBD	1.5	3000K	32256	1728	10494368	96	12	28672	720	BG728, CG717
XQ2V6000	TBD	1.5	6000K	76032	2592	21849504	144	12	67584	1104	CF1144

* Not for new designs

QPRO Radiation Tolerant

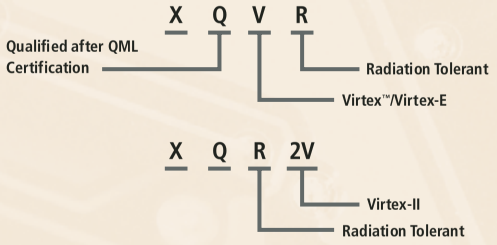
Device	Voltage	System Gates	Logic Cells	RAMbus	CFG Bits	MULTs	DLLs	Flip-Flops	Max I/Os	Packages	Total Ionizing Dose (TID) in KRADs
XQR4013XL	3.3	10-30K	1368	18K	393632	–	–	1536	192	CB228	60
XQR4036XL	3.3	22-65K	3078	42K	832528	–	–	3168	288	CB228	60
XQR4062XL	3.3	40-130K	5472	74K	1433864	–	–	5376	384	CB228	60
XQVR300	2.5	322970	6912	64K	1751840	–	4	6144	316	CB228	100
XQVR600	2.5	661111	15552	96K	3608000	–	4	13824	512	CB228	100
XQVR1000	2.5	1124022	27648	128K	6127776	–	4	24576	512	CG560	100
XQR2V1000	1.5	1000K	11520	720	4082592	40	8	10240	432	BG575	200
XQR2V3000	1.5	3000K	32256	1728	10494368	96	12	28672	720	CG717	200
XQR2V6000	1.5	6000K	76032	2592	21849504	144	12	67584	1104	CF1144	200

QPRO Configuration PROMs

Device	SMD	Voltage	System Gates	Logic Cells	RAMbus	CFG Bits	MULTs	DLLs	Flip-Flops	Max I/Os	Packages	Total Ionizing Dose (TID) in KRADs
XC1736D	None	5	–	–	–	32768	–	–	–	–	DD8	–
XC1765D	5962-94717	5	–	–	–	65536	–	–	–	–	DD8	–
XC17128D	None	5	–	–	–	131072	–	–	–	–	DD8	–
XC17256D	5962-95617	5	–	–	–	262144	–	–	–	–	DD8	–
XQ1701L	5962-99514	5	–	–	–	1048576	–	–	–	–	SO20, CC44	–
XQ17V16	TBD	3.3	–	–	–	16777216	–	–	–	–	VQ44, CC44	–
XQR1701L	–	3.3	–	–	–	1048576	–	–	–	–	SO20, CC44	60
XQR18V04	–	3.3	–	–	–	4194304	–	–	–	–	VQ44, CC44	40
XQR17V16**	–	3.3	–	–	–	16777216	–	–	–	–	VQ44, CC44	TBD

** Under development

Device Nomenclatures
XC = Qualified Prior to QML Certification



XILINX®
The Programmable Logic Company℠

Xilinx Global Services

http://www.xilinx.com/gsd/index.htm

Part Number	Product Description	Duration
Education Services		**Hours**
FPGA13000-6-ILT	Fundamentals of FPGA Design	8
FPGA23000-6-ILT	Designing for Performance V6	16
FPGA33000-6-ILT	Advanced FPGA Implementation	16
LANG11000-5-ILT	Introduction to VHDL	24
LANG21000-5-ILT	Advanced VHDL	16
LANG12000-5-ILT	Introduction to Verilog	24
PCI8000-4-ILT	PCI CORE Basics	8
PCI28000-4-ILT	Designing a PCI System	16
PCI2900-5-ILT	Designing for PCI-X	16
DSP2000-3-ILT	DSP Implementation Techniques for Xilinx FPGAs	24
DSP-100001-5-ILT	DSP Design Flow	24
RIO22000-6-ILT	Designing with Multi-Gigabit Serial I/O	16
EMBD-21000-5-ILT	Embedded Systems Development	16
Platinum Technical Service		**Hours**
SC-PLAT-SVC-10	Seat Platinum Technical Service w/10 education credits	N/A
SC-PLAT-SITE-50	Platinum Technical Service site license up to 50 customers	N/A
SC-PLAT-SITE-100	Platinum Technical Service site license for 51-100 customers	N/A
SC-PLAT-SITE-150	Platinum Technical Service site license for 101-150 customers	N/A
Titanium Technical Service		**Hours**
PS-TEC-SERV	Titanium Technical Service (minimum 40 hours)	N/A
DC-DES-SERV	Design Services Contract	N/A
Xilinx Productivity Advantage		**Hours**
DS-XPA	Custom XPA Packaged Solution	N/A
DS-XPA-10K	Custom XPA for $0 – $10,000	N/A
DS-XPA-25K	Custom XPA for $10,001 – $25,000	N/A
DS-XPA	Custom XPA for $25,001 – $100,000	N/A
DS-XPA-10K-INT	Custom XPA (International) for $0 – $10,000	N/A
DS-XPA-25K-INT	Custom XPA (International) for $10,001 – $25,000	N/A
DS-XPA-INT	Custom XPA (International) for $25,001 – $100,000	N/A
DS-ISE-ALI-XPA	XPA Seat, ISE Alliance	N/A
DS-ISE-FND-XPA	XPA Seat, ISE Foundation	N/A
Promotion Packages		**Hours**
PROMO-5003-6-ILT	Designing for Performance, Live Online	9
PROMO-5004-6-ILT	FPGA Essentials: Includes both Fundamentals of FPGA Design and Designing for Performance classes	24
DO-EDK-T	EDK Bundle (EDK + 10 TC's towards Embedded Systems Dev. Course)	NA
DS-SYSGEN-4SL-T	DSP Bundle (Sysgen + 15TC's towards DSP Design Flow Course)	NA

MySupport

- Sign up for personalized email alerts at mysupport.xilinx.com
- Search our knowledge database
- Troubleshoot with Problem Solvers
- Consult with engineers in Forums

Xilinx Productivity Advantage (XPA) Program

The Xilinx Productivity Advantage (XPA) delivers everything your customer needs to create their best design using Xilinx Programmable Solutions, including Software, Support, Services and IP cores. A single PO allows designers to get what they need, when they need it, at a great price.

Two types of XPAs are available. The custom XPA, tailored to your customers specific requirements, and the XPA Seat, an off-the-shelf pre-packaged solution for the individual designer.

http://support.xilinx.com/support/gsd/xpa_program.htm

Xilinx Design Services (XDS)

- XDS provides extensive FPGA hardware and embedded software design experience backed by industry recognized experts and resources to solve even the most complex design challenge.
- **System Architecture Consulting** — Provide engineering services to define system architecture and partitioning for design specification.
- **Custom Design Solutions** — Project designed, verified, and delivered to mutually agreed upon design specifications.
- **IP Core Development, Optimization, Integration, Modification, and Verification** — Modify, integrate, and optimize customer intellectual property or third party cores to work with Xilinx technology. Develop customer-required special features to Xilinx IP cores or third party cores. Perform integration, optimization, and verification of IP cores in Xilinx technology.
- **Embedded Software** — Develop complex embedded software with real-time constraints, using hardware/software co-design techniques.
- **Conversions** — Convert ASIC designs and other FPGAs to Xilinx technology and devices.

Titanium Technical Service

- Dedicated Application Engineer
- Service at customer site or at Xilinx
- Design flow methodology coaching
- Engineer/expertise specific to customer need
- All expertise levels available
- Service guaranteed via a contract

Platinum Technical Service

- Access to a dedicated team of Senior Applications Engineers
- Dedicated toll free number*
- Priority case resolution
- Ten Education Credits
- Electronic newsletter
- Formal escalation process
- Service Packs and Software updates
- Application Engineer to customer ratio is 2x the Gold Level

**Toll free number available in US only, dedicated local numbers available across Europe*

Education Services

- Reduce time to knowledge
- Improve design efficiency
- Reduce overall development costs

Public and private classes available worldwide

http://www.xilinx.com/support/education-home.htm

Xilinx Global Services
Finish Faster

Education Services Contacts

North America: 877-XLX-CLAS (877-959-2527)
http://support.xilinx.com/support/training/training.htm

Europe: +44-870-7350-548
eurotraining@xilinx.com

Japan: +81-3-5321-7750
http://support.xilinx.co.jp/support/education-home.htm

Asia Pacific: +852-2424-5200
education_ap@xilinx.com
http://support.xilinx.com/support/education-home.htm

Design Services Contacts

North America:
Richard Fodor: 408-626-4256
Mike Barone: 512-238-1473

Europe:
Alex Hillier: +44-870-7350-516
Martina Finnerty: +353-1-403-2469

Asia Pacific:
David Keefe: +852-2401-5171

designservices@xilinx.com

XPA Contacts

North America: 800-888-FPGA (3742)
fpga.xilinx.com

Europe:
Stuart Elston: +44-870-7350-532

Asia Pacific:
David Keefe: +852-2401-5171

Titanium Technical Service Contacts

North America:
Telesales: 800-888-3742

Europe:
Stuart Elston: +44-870-7350-632

Japan:
+81-3-5321-7730
or japantitanium@xilinx.com

Asia Pacific:
David Keefe: +852-2401-5171

XILINX®
The Programmable Logic Company℠

Xilinx Software – ISE 6.1i

http://www.xilinx.com/ise

	Feature	ISE WebPACK™	ISE BaseX	ISE Foundation	ISE Alliance
Devices	Virtex™ Series	**Virtex-E:** V50E -V300E **Virtex-II:** 2V40 - 2V250 **Virtex-II Pro:** 2VP2	**Virtex:** V50 - V600 **Virtex-E:** V50E - V600E **Virtex-II:** 2V40 - 2V500 **Virtex-II Pro:** 2VP2, 2VP4, 2VP7	ALL	ALL
	Spartan™ Series	**SpartanII/IIE:** ALL (except XC2S400E and XC2S600E) **Spartan-3:** 3S50, 3S200, 3S400	**Spartan-II/IIE:** ALL **Spartan-3:** 3S50, 3S200, 3S400	**Spartan-II/IIE:** ALL **Spartan-3:** ALL	**Spartan-II/IIE:** ALL **Spartan-3:** ALL
	CoolRunner™ XPLA3 CoolRunner-II	ALL	ALL	ALL	ALL
	XC9500 Series	ALL	ALL	ALL	ALL
Design Entry	Schematic Editor	Yes	MS Windows and Linux only	MS Windows and Linux only	MS Windows and Linux only
	HDL Editor	Yes	Yes	Yes	Yes
	State Diagram Editor	Yes	MS Windows only	MS Windows only	MS Windows only
	CORE Generator System	No	Yes	Yes	Yes
	PACE (Pinout and Area Constraint Editor)	Yes	Yes	Yes	Yes
	Architecture Wizards DCM - Digital Clock Management MGT - Multi-Gigabit Transcievers	Yes	Yes	Yes	Yes
	Third Party RTL Checker Support	Yes	Yes	Yes	Yes
	Xilinx System Generator for DSP	No	Sold as an Option	Sold as an Option	Sold as an Option
Embedded System Design	GNU Embedded Tools GCC - GNU Compiler GDB - GNU Software Debugger	No	Yes (Available with optional EDK)	Yes (Available with optional EDK)	Yes (Available with optional EDK)
	WindRiver Xilinx Edition Development Tools Diab C/C++ Compiler SingleStep Debugger visionPROBE II target connection	No	Sold as an Option	Sold as an Option	Sold as an Option
Synthesis	Xilinx Synthesis Technology (XST)	Yes	Yes	Yes	No
	Synplicity Synplify/Pro	Integrated Interface	Integrated Interface (MS Windows only)	Integrated Interface (MS Windows only)	Integrated Interface (MS Windows only)
	Synplicity Amplify Physical Synthesis Support	Yes	Yes	Yes	Yes
	Mentor Graphics Leonardo Spectrum	Integrated Interface	Integrated Interface	Integrated Interface	Integrated Interface
	Mentor Graphics Precision RTL	EDIF only	EDIF only	EDIF only	EDIF only
	Synopsys FPGA Compiler II	EDIF Interface	EDIF Interface	EDIF Interface	EDIF Interface
	ABEL	CPLD	CPLD (MS Windows only)	CPLD (MS Windows only)	CPLD (MS Windows only)

XILINX®
The Programmable Logic Company℠

Xilinx Software – ISE 6.1i

http://www.xilinx.com/ise

	Feature	*ISE WebPACK™*	*ISE BaseX*	*ISE Foundation*	*ISE Alliance*
Implementation	FloorPlanner	Yes	Yes	Yes	Yes
	Constraints Editor	Yes	Yes	Yes	Yes
	Timing Driven Place & Route	Yes	Yes	Yes	Yes
	Modular Design	No	Yes	Yes	Yes
	Timing Improvement Wizard	Yes	Yes	Yes	Yes
Programming	iMPACT	Yes	Yes	Yes	Yes
	System ACE Configuration Manager	Yes	Yes	Yes	Yes
Board Level Integration	IBIS Models	Yes	Yes	Yes	Yes
	STAMP Models	Yes	Yes	Yes	Yes
	HSPICE Models*	Yes	Yes	Yes	Yes
Verification	HDL Bencher™	Yes	MS Windows only	MS Windows only	MS Windows only
	ModelSim® Xilinx Edition (MXE II)	ModelSim XE II Starter**	ModelSim XE II Starter**	ModelSim XE II Starter**	ModelSim XE II Starter**
	Static Timing Analyzer	Yes	Yes	Yes	Yes
	ChipScope™ Pro	No	Sold as an Option	Sold as an Option	Sold as an Option
	FPGA Editor with Probe	No	Yes	Yes	Yes
	ChipViewer	Yes	Yes	Yes	Yes
	XPower (Power Analysis)	Yes	Yes	Yes	Yes
	Third Party Equivalency Checking Support	Yes	Yes	Yes	Yes
	SMARTModels for PPC and RocketIO	No	Yes	Yes	Yes
	Third Party Simulator Support	Yes	Yes	Yes	Yes
Platforms		PC (MS Windows 2000/MS Windows XP)	PC (MS Windows 2000/MS Windows XP), Linux	PC (MS Windows 2000/MS Windows XP), Sun, Solaris, Linux	PC(MS Windows 2000/MS Windows XP), Sun, Solaris, Linux
IP/CORE	For more information on the complete list of Xilinx IP products, visit the Xilinx IP Center at http://www.xilinx.com/ipcenter				

*HSPICE Models available at the Xilinx Design Tools Center at www.xilinx.com/ise.
**MXE II supports the simulation of designs up to 1 million system gates and is sold as an option.

For more information, visit the Xilinx Design Tools Center at www.xilinx.com/ise

Function	Vendor Name	IP Type	Virtex-II Pro	Virtex-II	Virtex-E	Virtex	Spartan-3	Spartan-IIE	Spartan-II	Spartan
Communication & Networking										
3G FEC Package	Xilinx	LogiCORE		V-II	V-E	V				
8b/10b Decoder	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
8b/10b Encoder	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
ADPCM, 1024 Channel Simplex (CS4190)	Amphion Semiconductor, Ltd.	AllianceCORE		V-II						
ADPCM, 256 Channel Simplex (CS4130)	Amphion Semiconductor, Ltd.	AllianceCORE		V-II	V-E					
ADPCM, 512 Channel Duplex (CS4180)	Amphion Semiconductor, Ltd.	AllianceCORE		V-II		V				
ADPCM, 768 Channel	Amphion Semiconductor, Ltd.	AllianceCORE		V-II		V				
AES Decryption Family (CS5200)	Amphion Semiconductor, Ltd.	AllianceCORE		V-II	V-E	V			S-II	
AES Encryption	CAST, Inc.	AllianceCORE	V-IIP	V-II	V-E		S-3	S-IIE		
AES Encryption Family (CS5200)	Amphion Semiconductor, Ltd.	AllianceCORE		V-II	V-E	V			S-II	
AES Standard Encryptor/Decryptor	Helion Technology Limited	AllianceCORE	V-IIP	V-II	V-E		S-3	S-IIE		
AES Tiny Encryptor/Decryptor	Helion Technology Limited	AllianceCORE	V-IIP	V-II	V-E		S-3	S-IIE		
ATM Adaption Layer 1 (AAL1)	ModelWare, Inc.	AllianceCORE		V-II	V-E					
ATM Cell Processor (CC200)	Paxonet Communications, Inc.	AllianceCORE	V-IIP	V-II						
AWGN - Additive White Gaussian Noise	Xilinx	LogiCORE	V-IIP	V-II						
Bit Stream Analyzer and Data Extractor (Parser)	Telecom Italia Lab S.p.A.	AllianceCORE	V-IIP	V-II				S-IIE		
Bluetooth Baseband Processor (BOOST Lite)	NewLogic GmbH	AllianceCORE		V-II		V				
CAM for Internet Protocol (IPlogiCAM)	Telecom Italia Lab S.p.A.	AllianceCORE	V-IIP	V-II				S-IIE		
Convolutional Encoder	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Convolutional Encoder (CONV_ENC)	Telecom Italia Lab S.p.A.	AllianceCORE	V-IIP	V-II				S-IIE		
CRC-32 for 10 Gbps OC192 Systems (CORE-CRC-128)	Calyptech Design Services	AllianceCORE					S-3			
CRC-32 for 40 Gbps OC-768 Systems (CORE-CRC-256)	Calyptech Design Services	AllianceCORE	V-IIP	V-II						
DES and DES3 Encryption Engine (MC-XIL-DES)	Memec Design	AllianceCORE		V-II	V-E	V		S-IIE	S-II	
DES Encryption	CAST, Inc.	AllianceCORE		V-II	V-E	V			S-II	
DES3 Encryption	CAST, Inc.	AllianceCORE	V-IIP	V-II	V-E		S-3	S-IIE		
Distributed Sample Descrambler (DSD)	Telecom Italia Lab S.p.A.	AllianceCORE				V			S-II	S
Distributed Sample Scrambler (DSS)	Telecom Italia Lab S.p.A.	AllianceCORE				V			S-II	S
DVB Satellite Modulator (MC-XIL-DVBMOD)	Memec Design	AllianceCORE		V-II	V-E	V			S-II	
Email Trigger	Amirix Systems, Inc.	AllianceCORE	V-IIP							
Ethernet 1000BASE-X PCS/PMA	Xilinx	LogiCORE	V-IIP							
Ethernet MAC, 1 Gigabit Full Duplex (PE-GMAC0)	Mentor Graphics Corporation	AllianceCORE		V-II						
Ethernet MAC, 1 Gigabit Half/Full Duplex with GMII or 1000BASE-X PCS/PMA	Xilinx	LogiCORE	V-IIP	V-II	V-E			S-IIE		
Ethernet MAC, 10 Gigabit Full Duplex with XGMII or XAUI	Xilinx	LogiCORE	V-IIP	V-II						
XAUI	Xilinx	LogiCORE	V-IIP							
Ethernet MAC, 10/100	Zuken, Inc.	AllianceCORE		V-II	V-E					
Ethernet MAC, 10/100 (MAC)	CAST, Inc.	AllianceCORE	V-IIP	V-II	V-E			S-IIE	S-II	
Ethernet MAC, 10/100 (PE-MACMII)	Mentor Graphics Corporation	AllianceCORE		V-II					S-II	
Ethernet MAC, 10G (CC410)	Paxonet Communications, Inc.	AllianceCORE		V-II						
Ethernet PCS, 10G (CC411)	Paxonet Communications, Inc.	AllianceCORE	V-IIP	V-II						
Ethernet PCS, 10G (MC-XIL-10GEPCS)	Memec Design	AllianceCORE		V-II						
Framer, 1.25 Gbps GFP (CC224)	Paxonet Communications, Inc.	AllianceCORE	V-IIP	V-II			S-3	S-IIE		
Framer, 2.5 Gbps GFP (CC226)	Paxonet Communications, Inc.	AllianceCORE	V-IIP	V-II						
Framer, 8-Bit Multilchannel GFP (CC225)	Paxonet Communications, Inc.	AllianceCORE	V-IIP	V-II						
Framer, 8-Bit Transparent GFP (CC124)	Paxonet Communications, Inc.	AllianceCORE	V-IIP	V-II						
Framer, E1 (CC303)	Paxonet Communications, Inc.	AllianceCORE	V-IIP	V-II				S-IIE		
Framer, OC12 (CC351)	Paxonet Communications, Inc.	AllianceCORE	V-IIP	V-II						
Framer, OC192/10 Gbps GFP (CC327)	Paxonet Communications, Inc.	AllianceCORE	V-IIP	V-II						
Framer, OTU2 (CC481)	Paxonet Communications, Inc.	AllianceCORE	V-IIP	V-II						
Framer, STS192/STM64 (CC314)	Paxonet Communications, Inc.	AllianceCORE	V-IIP	V-II						
Framer, T1 (CC302)	Paxonet Communications, Inc.	AllianceCORE	V-IIP	V-II				S-IIE		
Framer/Digital Wrapper, STS48 OTN (CC381)	Paxonet Communications, Inc.	AllianceCORE	V-IIP	V-II						
G.709 Compliant FEC Core (CC345)	Paxonet Communications, Inc.	AllianceCORE	V-IIP	V-II						
HDLC, Single-Channel	Memec Core	AllianceCORE	V-IIP	V-II			S-3			
HDLC, Single-Channel (MC-XIL-HDLC)	Memec Design	AllianceCORE	V-IIP	V-II			S-3			
HyperTransport Cave	GDA Technologies, Inc.	AllianceCORE	V-IIP	V-II						
Interleaver De-interleaver (INT_DEINT)	Telecom Italia Lab S.p.A.	AllianceCORE	V-IIP	V-II				S-IIE		
Interleaver/De-interleaver	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Inverse Multiplexer for ATM (IMA)	ModelWare, Inc.	AllianceCORE		V-II	V-E					
Mapper, E1 (CC333)	Paxonet Communications, Inc.	AllianceCORE	V-IIP	V-II				S-IIE		
MD5 Message Digest Algorithm	CAST, Inc.	AllianceCORE	V-IIP	V-II			S-3	S-IIE		
Modular Exponentiation Accelerator (MODEXP1)	CAST, Inc.	AllianceCORE	V-IIP	V-II	V-E		S-3	S-IIE		
Noisy Transmission Channel Model (CHANNEL)	Telecom Italia Lab S.p.A.	AllianceCORE	V-IIP	V-II				S-IIE		
Path Processor, OC12c (CC321)	Paxonet Communications, Inc.	AllianceCORE	V-IIP	V-II						
Path Processor, STS192/STM64 (CC324)	Paxonet Communications, Inc.	AllianceCORE	V-IIP	V-II						
Reed Solomon Decoder	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	S
Reed Solomon Decoder (MC-XIL-RSDEC)	Memec Design	AllianceCORE		V-II		V		S-IIE	S-II	
Reed Solomon Encoder	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	S
Reed Solomon Encoder (MC-XIL-RSENC)	Memec Design	AllianceCORE		V-II		V		S-IIE	S-II	
Reed-Solomon Decoder (RS_DEC)	Telecom Italia Lab S.p.A.	AllianceCORE	V-IIP	V-II				S-IIE		
SDLC Controller (SDLC)	CAST, Inc.	AllianceCORE		V-II		V				
SHA-1 Encryption Processor	CAST, Inc.	AllianceCORE		V-II	V-E	V		S-IIE	S-II	
SPI 4.2 Interface (CC401)	Paxonet Communications, Inc.	AllianceCORE	V-IIP	V-II						
SPI-3 (POS-PHY L3) Link Layer Interface, Multi-Channel	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
SPI-3 (POS-PHY L3) Link Layer Interface, 1-Ch	Xilinx	LogiCORE	V-IIP	V-II	V-E	V		S-IIE	S-II	
SPI-3 (POS-PHY L3) Link Layer Interface, 2-Ch	Xilinx	LogiCORE	V-IIP	V-II	V-E	V		S-IIE	S-II	
SPI-3 (POS-PHY L3) Link Layer Interface, 4-Ch	Xilinx	LogiCORE	V-IIP	V-II	V-E	V		S-IIE	S-II	
SPI-3 (POS-PHY L3) Physical Layer Interface	Xilinx	LogiCORE			V-E					
SPI-4.1 (Flexbus 4) Interface Core, 1-Channel	Xilinx	LogiCORE		V-II						
SPI-4.1 (Flexbus 4) Interface Core, 4-Channel	Xilinx	LogiCORE		V-II						
SPI-4.2 (POS-PHY L4) Multi-Channel Interface	Xilinx	LogiCORE	V-IIP	V-II						
SPI-4.2 (POS-PHY L4) to SPI-4.1 (Flexbus 4) Bridge	Xilinx	LogiCORE		V-II						
SPI-4.2 (POS-PHY L4) to XGMII (10GE MAC) Bridge	Xilinx	LogiCORE		V-II						

Visit the Xilinx IP Center for more details at *www.xilinx.com/ipcenter*

Function	Vendor Name	IP Type	Virtex-II Pro	Virtex-II	Virtex-E	Virtex	Spartan-3	Spartan-IIE	Spartan-II	Spartan
SPI-4.2 Lite (POS_PHY L4)	Xilinx	LogiCORE		V-II			S-3			
Turbo Decoder (TURBO_DEC)	Telecom Italia Lab S.p.A.	AllianceCORE		V-II		V				
Turbo Decoder, 3GPP	SysOnChip, Inc.	AllianceCORE		V-II	V-E					
Turbo Decoder, 3GPP (S3000)	iCoding Technology, Inc.	AllianceCORE		V-II	V-E			S-IIE		
Turbo Decoder, Convolutional, 3GPP Compliant	Xilinx	LogiCORE		V-II	V-E	V				
Turbo Decoder, Convolutional, 3GPP2/CDMA2000	Xilinx	LogiCORE	V-IIP	V-II			S-3			
Turbo Decoder, DVB-RCS (S2000)	iCoding Technology, Inc.	AllianceCORE		V-II	V-E	V				
Turbo Decoder, DVB-RCS (TC1000)	TurboConcept	AllianceCORE		V-II	V-E					
Turbo Decoder, Product Code	Xilinx	LogiCORE	V-IIP	V-II			S-3			
Turbo Encoder (TURBO_ENC)	Telecom Italia Lab S.p.A.	AllianceCORE	V-IIP	V-II				S-IIE		
Turbo Encoder, Convolutional, 3GPP Compliant	Xilinx	LogiCORE		V-II	V-E	V				
Turbo Encoder, Convolutional, 3GPP2/CDMA2000	Xilinx	LogiCORE	V-IIP	V-II			S-3			
Turbo Encoder, DVB-RCS (S2001)	iCoding Technology, Inc.	AllianceCORE		V-II	V-E	V				
Turbo Encoder, Product Code	Xilinx	LogiCORE	V-IIP	V-II			S-3			
Turbo Product Code Decoder, 25 Mbps (TC3000)	TurboConcept	AllianceCORE		V-II	V-E					
Turbo Product Code Decoder, 30 Mbps (TC3401)	TurboConcept	AllianceCORE			V-E			S-IIE		
Turbo Product Code Decoder, 160 Mbps (TC3404)	TurboConcept	AllianceCORE		V-II	V-E					
UTOPIA Level-2 PHY Side RX Interface (UTOPIA L2 PHY Rx)	Telecom Italia Lab S.p.A.	AllianceCORE	V-IIP	V-II				S-IIE		
UTOPIA Level-2 PHY Side TX Interface (UTOPIA L2 PHY Tx)	Telecom Italia Lab S.p.A.	AllianceCORE	V-IIP	V-II				S-IIE		
UTOPIA RX Level 2 Master Interface (UTOPIA2M_RX)	Telecom Italia Lab S.p.A.	AllianceCORE	V-IIP	V-II				S-IIE		
Viterbi Decoder (VITERBI_DEC)	Telecom Italia Lab S.p.A.	AllianceCORE	V-IIP	V-II				S-IIE		
Viterbi Decoder, General Purpose	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Viterbi Decoder, IEEE 802-Compatible	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
XAPP 289: Common Switch Interface CSIX-L1 Reference Design	Xilinx	Reference Design	V-IIP	V-II						
Digital Signal Processing										
Bit Correlator	Xilinx	LogiCORE	V-IIP	V-II	V-E	V		S-IIE	S-II	
Cascaded Integrator Comb (CIC) Filter	Xilinx	LogiCORE	V-IIP	V-II	V-E	V		S-IIE	S-II	
Comb Filter	Xilinx	LogiCORE								S
CORDIC	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Digital Down Converter (DDC)	Xilinx	LogiCORE	V-IIP	V-II	V-E	V		S-IIE	S-II	
Digital Up Converter (DUC)	Xilinx	LogiCORE	V-IIP							
Direct Digital Synthesizer (DDS)	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Discrete Cosine Transform (eDCT)	eInfochips Pvt. Ltd.	AllianceCORE		V-II	V-E	V		S-IIE	S-II	
Discrete Cosine Transform, 2D Inverse (IDCT)	CAST, Inc.	AllianceCORE		V-II	V-E	V			S-II	
Discrete Cosine Transform, Combined 2D Forward/Inverse (DCT_FI)	CAST, Inc.	AllianceCORE		V-II	V-E	V			S-II	
Discrete Cosine Transform, Forward 2D (DCT)	CAST, Inc.	AllianceCORE		V-II	V-E	V			S-II	
Discrete Cosine Transform, Forward/Inverse (FIDCT)	Telecom Italia Lab S.p.A.	AllianceCORE	V-IIP	V-II				S-IIE		
Discrete Cosine Transform, Forward/Inverse 2D (DCT/IDCT 2D)	Barco-Silex	AllianceCORE		V-II	V-E	V			S-II	
Discrete Wavelet Transform, Combined 2D Forward/Inverse (RC_2DDWT)	CAST, Inc.	AllianceCORE		V-II	V-E	V			S-II	
Discrete Wavelet Transform, Line-Based Programmable Forward (LB_2DFDWT)	CAST, Inc.	AllianceCORE		V-II	V-E	V			S-II	
DOCSIS ITU-T J.83 Modulator	Xilinx	LogiCORE	V-IIP	V-II			S-3			
DOCSIS ITU-T J.83 Modulator	Xilinx	LogiCORE	V-IIP	V-II			S-3			
DSP Hardware Accelerator, Virtex-E (GVA-290)	GV & Associates, Inc.	AllianceCORE			V-E					
DSP Hardware Accelerator, Virtex-II (GVA-300)	GV & Associates, Inc.	AllianceCORE		V-II						
DSP Hardware Accelerator, Virtex-II (GVA-325)	GV & Associates, Inc.	AllianceCORE		V-II						
DSP Hardware Accelerator, Virtex-II (GVA-350)	GV & Associates, Inc.	AllianceCORE		V-II						
Fast Fourier Transform	Xilinx	LogiCORE	V-IIP	V-II			S-3			
FFT/IFFT for Virtex-II, 1024-Point Complex	Xilinx	LogiCORE		V-II						
FFT/IFFT for Virtex-II, 16-Point Complex	Xilinx	LogiCORE		V-II						
FFT/IFFT for Virtex-II, 256-Point Complex	Xilinx	LogiCORE		V-II						
FFT/IFFT for Virtex-II, 64-Point Complex	Xilinx	LogiCORE		V-II						
FFT/IFFT, 1024-Point Complex	Xilinx	LogiCORE			V-E	V				
FFT/IFFT, 16-Point Complex	Xilinx	LogiCORE		V-II	V-E	V				
FFT/IFFT, 256-Point Complex	Xilinx	LogiCORE		V-II	V-E	V				
FFT/IFFT, 32-Point Complex	Xilinx	LogiCORE	V-IIP	V-II	V-E	V		S-IIE	S-II	
FFT/IFFT, 64-, 256-, 1024-Point Complex	Xilinx	LogiCORE	V-IIP	V-II			S-3			
FFT/IFFT, 64-Point Complex	Xilinx	LogiCORE			V-E	V				
FIR Filter Using DPRAM	eInfochips Pvt. Ltd.	AllianceCORE		V-II	V-E	V		S-IIE	S-II	
FIR Filter, Distributed Arithmetic (DA)	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
FIR Filter, Distributed Arithmetic Parallel	Xilinx	LogiCORE								S
FIR Filter, Distributed Arithmetic Serial	Xilinx	LogiCORE								S
FIR Filter, MAC	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
FIR Filter, Parallel Distributed Arithmetic	eInfochips Pvt. Ltd.	AllianceCORE		V-II	V-E	V		S-IIE	S-II	
LFSR, Linear Feedback Shift Register	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Oscillator, Dual-Channel Numerically Controlled	Xilinx	LogiCORE			V-E	V		S-IIE	S-II	S
Oscillator, Numerically Controlled	Xilinx	LogiCORE			V-E	V		S-IIE	S-II	S
Time-Skew Buffer, Nonsymmetric 16-Deep	Xilinx	LogiCORE								S
Time-Skew Buffer, Nonsymmetric 32-Deep	Xilinx	LogiCORE								S
Time-Skew Buffer, Symmetric 16-Deep	Xilinx	LogiCORE								S
TMS32025 DSP Processor (C32025)	CAST, Inc.	AllianceCORE		V-II	V-E	V			S-II	
Math Functions										
1s and 2s Complement	Xilinx	LogiCORE								S
Accumulator	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Adder Subtracter	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Divider, Pipelined	Xilinx	LogiCORE		V-II	V-E	V		S-IIE	S-II	S
Floating Point Adder (DFPADD)	Digital Core Design	AllianceCORE		V-II		V			S-II	
Floating Point Comparator (DFPCOMP)	Digital Core Design	AllianceCORE		V-II		V			S-II	
Floating Point Divider (DFPDIV)	Digital Core Design	AllianceCORE		V-II		V			S-II	
Floating Point Multiplier (DFPMUL)	Digital Core Design	AllianceCORE		V-II	V-E	V			S-II	
Floating Point Square Root Operator (DFPSQRT)	Digital Core Design	AllianceCORE		V-II	V-E				S-II	
Floating Point to Integer Converter (DFP2INT)	Digital Core Design	AllianceCORE		V-II		V			S-II	
Integer to Floating Point Converter (DINT2FP)	Digital Core Design	AllianceCORE		V-II		V			S-II	
Integrator	Xilinx	LogiCORE								S

Visit the Xilinx IP Center for more details at *www.xilinx.com/ipcenter*

Function	Vendor Name	IP Type	Virtex-II Pro	Virtex-II	Virtex-E	Virtex	Spartan-3	Spartan-IIE	Spartan-II	Spartan
Multiplier for Virtex, Variable Parallel	Xilinx	LogiCORE			V-E	V		S-IIE	S-II	
Multiplier, Constant Coefficient	Xilinx	LogiCORE								S
Multiplier, Constant Coefficient - Pipelined	Xilinx	LogiCORE								S
Multiplier, Dynamic Constant Coefficient	Xilinx	LogiCORE			V-E	V				
Multiplier, Parallel - Area Optimized	Xilinx	LogiCORE								S
Multiply Accumulator (MAC)	Xilinx	LogiCORE	V-IIP	V-II		V	S-3	S-IIE	S-II	
Multiply Generator	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Registered Adder	Xilinx	LogiCORE								S
Registered Loadable Adder	Xilinx	LogiCORE								S
Registered Loadable Subtracter	Xilinx	LogiCORE								S
Registered Scaled Adder	Xilinx	LogiCORE								S
Registered Serial Adder	Xilinx	LogiCORE								S
Registered Subtracter	Xilinx	LogiCORE								S
Scaled-by-One-Half Accumulator	Xilinx	LogiCORE								S
Sine Cosine Look Up Table	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	S
Square Root	Xilinx	LogiCORE								S
Twos Complementer	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Memories & Storage Elements										
ATM Utopia Level 2	Xilinx	LogiCORE	V-IIP							
Block Memory, Dual-Port	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Block Memory, Single-Port	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Content Addressable Memory (CAM)	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Distributed Memory	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
FIFO, Asynchronous	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
FIFO, Synchronous	Xilinx	LogiCORE								S
FIFO, Synchronous	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Pipelined Delay Element	Xilinx	LogiCORE								S
Registered ROM	Xilinx	LogiCORE								S
Registered Single Port RAM	Xilinx	LogiCORE								S
RLDRAM Memory Controller	Avnet Design Services	AllianceCORE		V-II						
SDRAM Controller (EP520)	Eureka Technology	AllianceCORE				V				
SDRAM Controller, DDR (EP525)	Eureka Technology	AllianceCORE	V-IIP	V-II			S-3	S-IIE		
SDRAM Controller, DDR (MC-XIL-SDRAMDDR)	Memec Design	AllianceCORE		V-II	V-E				S-II	
Microprocessors, Controllers & Peripherals										
1 Gigabit Ethernet MAC w/PLB Interface	Xilinx	LogiCORE	V-IIP							
10/100 Ethernet MAC Lite w/OPB Interface	Xilinx	LogiCORE	V-IIP	V-II	V-E	V		S-IIE	S-II	
10/100 Ethernet MAC w/OPB Interface	Xilinx	LogiCORE	V-IIP	V-II	V-E	V		S-IIE	S-II	
16450 UART (H16450)	CAST, Inc.	AllianceCORE		V-II	V-E	V		S-IIE	S-II	
16450 UART w/OPB Interface	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
16450 UART w/Synchronous Interface (H16450S)	CAST, Inc.	AllianceCORE		V-II	V-E	V		S-IIE	S-II	
16550 UART w/FIFOs & Synchronous Interface (H16550S)	CAST, Inc.	AllianceCORE		V-II	V-E	V		S-IIE	S-II	
16550 UART w/FIFOs (H16550)	CAST, Inc.	AllianceCORE		V-II	V-E	V		S-IIE	S-II	
16550 UART w/OPB interface	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
2910A Microprogram Controller (C2910A)	CAST, Inc.	AllianceCORE				V			S-II	S
2D Fabric Evaluation and Demo Board	Crossbow Technologies, Inc.	AllianceCORE							S-II	
2D Multiprocessing Interface Fabric (2D-fabric402C)	Crossbow Technologies, Inc.	AllianceCORE	V-IIP	V-II				S-IIE	S-II	
68000 Compatible Microprocessor (C68000)	CAST, Inc.	AllianceCORE		V-II	V-E	V	S-3			
80186 Compatible Microprocessor (e80186)	eInfochips Pvt. Ltd.	AllianceCORE		V-II		V				
8051 Base Compatible Microcontroller (DR8051BASE)	Digital Core Design	AllianceCORE		V-II		V			S-II	
8051 Compatible Microcontroller (C8051)	CAST, Inc.	AllianceCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
8051 Compatible Microcontroller (Flip805x-PR)	Dolphin Integration	AllianceCORE				V			S-II	S
8051 RISC Microcontroller (DR8051)	Digital Core Design	AllianceCORE		V-II		V			S-II	
80515 High-speed 8-bit RISC Microcontroller (R80515)	CAST, Inc.	AllianceCORE				V				
8052 Compatible Microcontroller (DR8052EX)	Digital Core Design	AllianceCORE		V-II		V			S-II	
80C51 Compatible RISC Microcontroller (R8051)	CAST, Inc.	AllianceCORE			V-E	V			S-II	
8237 Programmable DMA Controller (C8237)	CAST, Inc.	AllianceCORE		V-II	V-E	V			S-II	
8250 UART (H8250)	CAST, Inc.	AllianceCORE		V-II	V-E			S-IIE	S-II	
8254 Programmable Interval Timer/Counter (C8254)	CAST, Inc.	AllianceCORE		V-II	V-E	V			S-II	
8254 Programmable Interval Timer/Counter (e8254)	eInfochips Pvt. Ltd.	AllianceCORE		V-II					S-II	
8255 Programmable I/O Controller (e8255)	eInfochips Pvt. Ltd.	AllianceCORE		V-II					S-II	
8259A Programmable Interrupt Controller (C8259A)	CAST, Inc.	AllianceCORE		V-II	V-E	V			S-II	
Arbiter and Bus Structure w/OPB Interface	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Arbiter and Bus Structure w/PLB Interface	Xilinx	LogiCORE	V-IIP							
ATM Utopia Level 2 Master and Slave w/OPB Interface	Xilinx	LogiCORE	V-IIP	V-II	V-E	V		S-IIE	S-II	
ATM Utopia Level 2 Master and Slave w/PLB Interface	Xilinx	LogiCORE	V-IIP							
BRAM Controller w/LMB Interface	Xilinx	LogiCORE	V-IIP	V-II	V-E	V		S-IIE	S-II	
BRAM Controller w/OPB Interface	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
BRAM Controller w/PLB Interface	Xilinx	LogiCORE	V-IIP							
BSP Generator (SW only)	Xilinx	LogiCORE	V-IIP							
Compact Video Controller (logiCVC)	Xylon d.o.o.	AllianceCORE		V-II		V			S-II	
CRT Controller (C6845)	CAST, Inc.	AllianceCORE		V-II	V-E			S-IIE	S-II	
DCR Bus Structure	Xilinx	LogiCORE	V-IIP							
External Memory Controller (EMC) w/OPB Interface (Includes support for Flash, SRAM, ZBT, System ACE)	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
External Memory Controller (EMC) w/PLB Interface (Includes support for Flash, SRAM, ZBT, System ACE)	Xilinx	LogiCORE	V-IIP							
FPU for MicroBlaze	QinetiQ Limited	AllianceCORE	V-IIP	V-II						
Generic compact UART	Memec Core	Submitted	V-IIP	V-II			S-3			
GPIO w/OPB Interface	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
HDLC Controller (Single Channel) w/OPB Interface	Xilinx	LogiCORE	V-IIP	V-II	V-E	V		S-IIE	S-II	
IIC w/OPB Interface	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Internet Appliance (socPiP-1A_Platform)	SoC Solutions, LLC	AllianceCORE		V-II	V-E					
Interrupt Controller (IntC) w/DCR Interface	Xilinx	LogiCORE	V-IIP	V-II	V-E	V		S-IIE	S-II	

Visit the Xilinx IP Center for more details at *www.xilinx.com/ipcenter*

Function	Vendor Name	IP Type	Virtex-II Pro	Virtex-II	Virtex-E	Virtex	Spartan-3	Spartan-IIE	Spartan-II	Spartan
Interrupt Controller (IntC) w/OPB Interface	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
IPIF Address Decode w/OPB Interface	Xilinx	LogiCORE	V-IIP	V-II	V-E	V		S-IIE	S-II	
IPIF DMA w/OPB Interface	Xilinx	LogiCORE	V-IIP	V-II	V-E	V		S-IIE	S-II	
IPIF Interrupt Controller w/OPB Interface	Xilinx	LogiCORE	V-IIP	V-II	V-E	V		S-IIE	S-II	
IPIF Master/Slave Attachment w/OPB Interface	Xilinx	LogiCORE	V-IIP	V-II	V-E	V		S-IIE	S-II	
IPIF Read/Write Packet FIFO w/OPB Interface	Xilinx	LogiCORE	V-IIP	V-II	V-E	V		S-IIE	S-II	
IPIF Scatter/Gather w/OPB Interface	Xilinx	LogiCORE	V-IIP	V-II	V-E	V		S-IIE	S-II	
IPIF Slave Attachment w/PLB Interface	Xilinx	LogiCORE	V-IIP							
Java Processor, 32-bit (Lightfoot)	Digital Communications Technologies, Ltd.	AllianceCORE		V-II		V			S-II	
Java Processor, Configurable (LavaCORE)	Derivation Systems, Inc.	AllianceCORE		V-II		V				
JTAG UART w/OPB Interface	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Memory Test Utility (SW only)	Xilinx	LogiCORE	V-IIP							
MicroBlaze Soft RISC Processor	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
MicroBlaze Source Code	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
MIPS System Controller (ES500)	Eureka Technology	AllianceCORE		V-II	V-E					
ML300 VxWorks BSP (SW only)	Xilinx	LogiCORE	V-IIP							
Motor Controller - 3 Phase (MLCA_4)	MEET Ltd.	AllianceCORE						S-IIE	S-II	S
OPB2DCR Bridge	Xilinx	LogiCORE	V-IIP							
OPB2OPB Bridge (Lite)	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
OPB2PCI Full Bridge (32/33)	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
OPB2PLB Bridge	Xilinx	LogiCORE	V-IIP							
Operating System Accelerator (Sierra S16)	RealFast Operating Systems AB	AllianceCORE		V-II				S-IIE	S-II	
PC/104-Plus Reconfigurable Module Board (PF3100)	Derivation Systems, Inc.	AllianceCORE								
PIC125x Fast RISC Microcontroller (DFPIC125X)	Digital Core Design	AllianceCORE		V-II		V			S-II	
PIC1655x Fast RISC Microcontroller (DFPIC1655X)	Digital Core Design	AllianceCORE		V-II		V			S-II	
PIC165X Compatible Microcontroller (C165X)	CAST, Inc.	AllianceCORE		V-II	V-E	V			S-II	
PIC165x Fast RISC Microcontroller (DFPIC165X)	Digital Core Design	AllianceCORE		V-II	V-E	V			S-II	
PIC16C55X Compatible RISC Microcontroller (C1655x)	CAST, Inc.	AllianceCORE		V-II	V-E	V		S-IIE	S-II	
PicoBlaze (XAPP 213: PicoBlaze 8-bit Microcontroller for Virtex-E and Spartan-II/E Devices)	Xilinx	Reference Design			V-E	V		S-IIE	S-II	
PicoBlaze (XAPP 627: PicoBlaze 8-bit Microcontroller for Virtex-II and Virtex-II Pro Devices)	Xilinx	Reference Design	V-IIP	V-II						
PicoBlaze Emulated 8051 Microcontroller (PB8051-MX/TF)	Roman-Jones, Inc.	AllianceCORE	V-IIP	V-II			S-3	S-IIE		
PLB2OPB Bridge	Xilinx	LogiCORE	V-IIP							
PowerPC Bus Master (EP201)	Eureka Technology	AllianceCORE	V-IIP	V-II			S-3	S-IIE		
PowerPC Bus Slave (EP100)	Eureka Technology	AllianceCORE	V-IIP	V-II			S-3	S-IIE		
RISC Processor, 16-bit Proprietary (AX1610)	Loarant Corporation	AllianceCORE	V-IIP	V-II			S-3	S-IIE		
SDRAM Controller w/OPB Interface	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
SDRAM Controller w/PLB Interface	Xilinx	LogiCORE	V-IIP							
SPI Master and Slave w/OPB Interface	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Systems Reset Module	Xilinx	LogiCORE	V-IIP							
Timebase/Watch Dog Timer (WDT) w/OPB Interface	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Timer/Counter w/OPB Interface	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
UART Lite w/OPB Interface	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
UART, Generic Compact (MC-XIL-UART)	Memec Design	AllianceCORE	V-IIP	V-II			S-3			
UltraContoller Solution: A Lightweight PowerPC Microcontroller (XAPP672)	Xilinx	Reference Design	V-IIP							
VxWorks Board Support Package (BSP)	Xilinx	LogiCORE	V-IIP							
XTENSA-V Configurable 32-bit Microprocessor	Tensilica, Inc.	AllianceCORE		V-II						
Z80 Compatible Microprocessor (CZ80CPU)	CAST, Inc.	AllianceCORE		V-II	V-E	V	S-3		S-II	
Z80 Compatible Programmable Counter/Timer (CZ80CTC)	CAST, Inc.	AllianceCORE		V-II	V-E	V			S-II	
Z80 Peripheral I/O Controller (CZ80PIO)	CAST, Inc.	AllianceCORE		V-II	V-E	V		S-IIE	S-II	
Standard Bus Interfaces										
Arbiter	Telecom Italia Lab S.p.A.	AllianceCORE	V-IIP	V-II				S-IIE		
CAN 2.0 B Compatible Network Controller (LogiCAN)	Xylon d.o.o.	AllianceCORE	V-IIP	V-II			S-3	S-IIE		
CAN Bus Controller (MC-XIL-OPB-XCAN)	Memec Design	AllianceCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
CAN Bus Controller 2.0B	CAST, Inc.	AllianceCORE	V-IIP	V-II			S-3	S-IIE		
CAN Bus Controller with 32 Mail Boxes	Robert Bosch GmbH	AllianceCORE	V-IIP	V-II			S-3	S-IIE		
HyperTransport Cave, 8-bit	GDA Technologies, Inc.	AllianceCORE	V-IIP	V-II						
HyperTransport Single-Ended Slave Core	Xilinx	LogiCORE	V-IIP	V-II						
I2C Bus Controller (I2C)	CAST, Inc.	AllianceCORE		V-II	V-E	V	S-3		S-II	
I2C Bus Controller Master (DI2CM)	Digital Core Design	AllianceCORE		V-II	V-E				S-II	S
I2C Bus Controller Slave (DI2CS)	Digital Core Design	AllianceCORE		V-II	V-E				S-II	S
I2C Bus Controller Slave Base (DI2CSB)	Digital Core Design	AllianceCORE		V-II	V-E				S-II	S
I2C Two-Wire Serial Interface Master-Only (MC-XIL-TWSIMO)	Memec Design	AllianceCORE		V-II	V-E	V		S-IIE	S-II	
I2C Two-Wire Serial Interface Master-Slave (MC-XIL-TWSIMS)	Memec Design	AllianceCORE		V-II	V-E	V		S-IIE	S-II	
LIN - Local Interconnect Network Bus Controller (iLIN)	Intelliga Integrated Design, Ltd.	AllianceCORE	V-IIP	V-II	V-E		S-3	S-IIE		
PCI 64-bit/66-MHz Master/Target Interface (EC240)	Eureka Technology	AllianceCORE		V-II	V-E					
PCI Express Endpoint Core	Xilinx	LogiCORE	V-IIP							
PCI Host Bridge (EP430)	Eureka Technology	AllianceCORE		V-II	V-E					
PCI, 64-bit Target Interface (PCI-T64)	CAST, Inc.	AllianceCORE		V-II	V-E			S-IIE		
PCI32 Interface Design Kit (DO-DI-PCI32-DKT)	Xilinx	LogiCORE	V-IIP	V-II	V-E	V		S-IIE	S-II	
PCI32 Interface, IP Only (DO-DI-PCI32-IP)	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
PCI32 Single-Use License for Spartan (DO-DI-PCI32-SP)	Xilinx	LogiCORE					S-3	S-IIE	S-II	
PCI64 & PCI32, IP Only (DO-DI-PCI-AL)	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
PCI64 Interface Design Kit (DO-DI-PCI64-DKT)	Xilinx	LogiCORE	V-IIP	V-II	V-E	V		S-IIE	S-II	
PCI64 Interface, IP Only (DO-DI-PCI64-IP)	Xilinx	LogiCORE	V-IIP	V-II	V-E	V		S-IIE	S-II	
PCI-X 64/133 Interface for Virtex-II (DO-DI-PCIX64-VE) Includes PCI 64 bit interface at 33 MHz	Xilinx	LogiCORE	V-IIP	V-II						
PCI-X 64/66 Interface for Virtex-E (DO-DI-PCIX64-VE) Includes PCI 64 bit interface at 33 MHz	Xilinx	LogiCORE			VE					
RapidIO 8-bit port LP-LVDS Phy Layer (DO-DI-RIO8-PHY)	Xilinx	LogiCORE	V-IIP	V-II						
RapidIO Logical (I/O) and Transport Layer (DO-DI-RIO8-LOG)	Xilinx	LogiCORE	V-IIP	V-II						
RapidIO Phy Layer to PLB Bridge Reference Design	Xilinx	Reference Design	V-IIP							

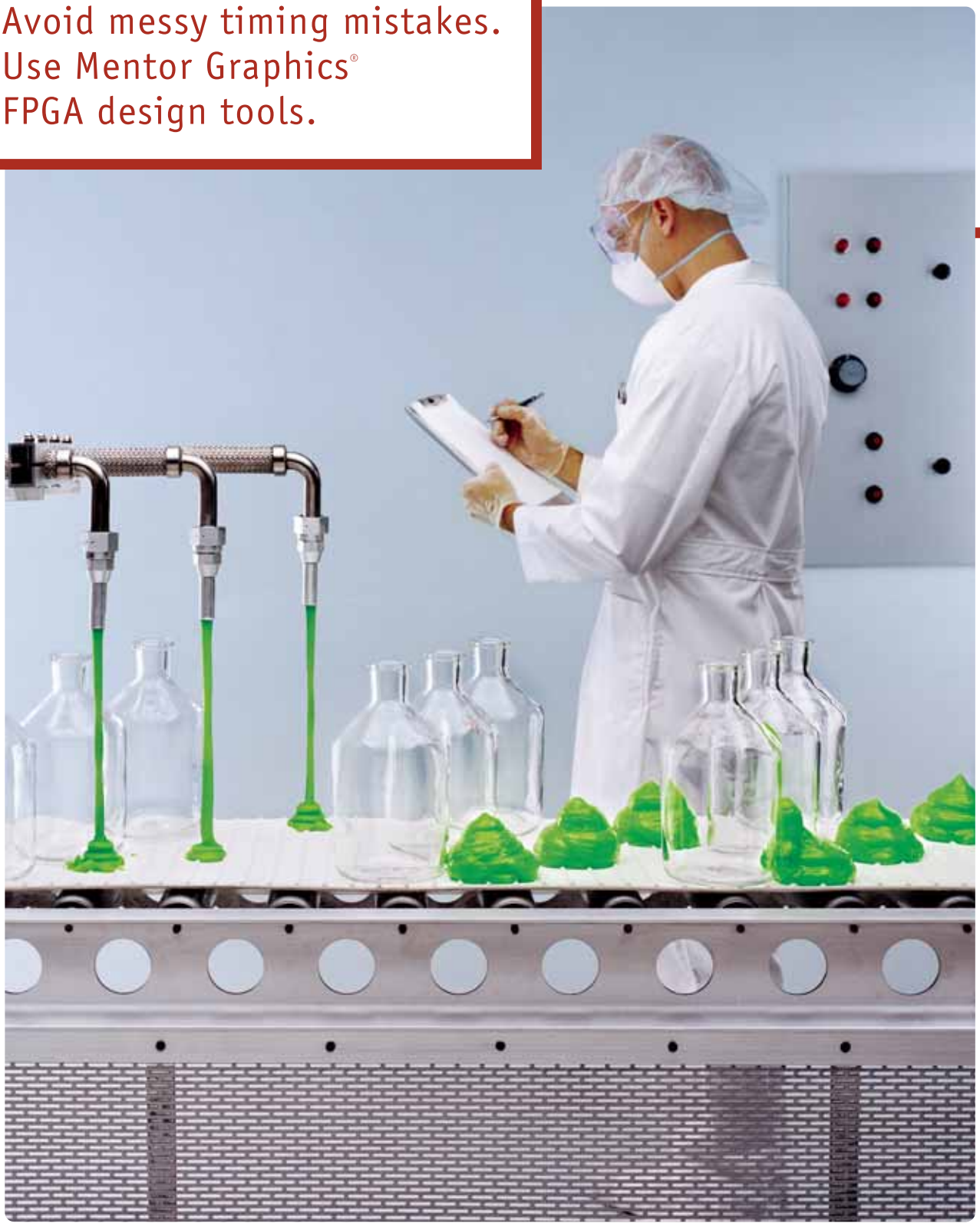
Visit the Xilinx IP Center for more details at *www.xilinx.com/ipcenter*

Function	Vendor Name	IP Type	Virtex-II Pro	Virtex-II	Virtex-E	Virtex	Spartan-3	Spartan-IIE	Spartan-II	Spartan
Serial Protocol Interface Slave (SPI Slave)	CAST, Inc.	AllianceCORE		V-II	V-E	V		S-IIE		
USB 1.1 Function Controller (CUSB)	CAST, Inc.	AllianceCORE		V-II	V-E	V			S-II	
USB 2.0 Function Controller (CUSB2)	CAST, Inc.	AllianceCORE	V-IIP	V-II	V-E		S-3	S-IIE		
XAPP653: Virtex-II Pro/Spartan-3 3.3V PCI Reference Design	Xilinx	Reference Design	V-IIP				S-3			
Video & Image Processing										
Burst Locked PLL (BURST_PLL)	Pinpoint Solutions, Inc.	AllianceCORE		V-II	V-E		S-3	S-IIE		
Color Space Converter, RGB2YCrCb	Perigee, LLC	AllianceCORE				V			S-II	S
Color Space Converter, RGB2YCrCb (CSC)	CAST, Inc.	AllianceCORE		V-II	V-E	V		S-IIE	S-II	
Color Space Converter, YCrCb2RGB	Perigee, LLC	AllianceCORE				V			S-II	S
Huffman Decoder (HUFFD)	CAST, Inc.	AllianceCORE		V-II	V-E	V		S-IIE	S-II	
JPEG Fast Codec (JPEG_FAST_C)	CAST, Inc.	AllianceCORE	V-IIP	V-II	V-E	V	S-3			
JPEG, 2000 Encoder (JPEG2K_E)	CAST, Inc.	AllianceCORE	V-IIP	V-II	V-E					
JPEG, Fast Color Image Decoder (FASTJPEG_C DECODER)	Barco-Silex	AllianceCORE		V-II	V-E	V				
JPEG, Fast Decoder (JPEG_FAST_D)	CAST, Inc.	AllianceCORE	V-IIP	V-II	V-E	V		S-IIE		
JPEG, Fast Encoder (JPEG_FAST_E)	CAST, Inc.	AllianceCORE	V-IIP	V-II	V-E	V				
JPEG, Fast Grayscale Image Decoder (FASTJPEG_BW DECODER)	Barco-Silex	AllianceCORE		V-II	V-E	V				
JPEG, Motion Codec V1.0 (CS6190)	Amphion Semiconductor, Ltd.	AllianceCORE		V-II	V-E	V				
JPEG, Motion Decoder (CS6150)	Amphion Semiconductor, Ltd.	AllianceCORE		V-II	V-E	V				
JPEG, Motion Encoder (CS6100)	Amphion Semiconductor, Ltd.	AllianceCORE		V-II	V-E	V				
MPEG-2 HDTV I & P Encoder (DV1 HDTV)	Duma Video, Inc.	AllianceCORE		V-II						
MPEG-2 SDTV I & P Encoder (DV1 SDTV)	Duma Video, Inc.	AllianceCORE		V-II						
NTSC Color Separator (NTSC-COSEP)	Pinpoint Solutions, Inc.	AllianceCORE		V-II	V-E		S-3	S-IIE		
Backplanes and Gigabit Serial I/O										
Aurora 201, 401and 804 Designs	Xilinx	Reference Design	V-IIP							
WP160: Emulating External SERDES Devices with Embedded RocketIO Transceivers	Xilinx	White Paper	V-IIP							
XAPP 649: SONET Rate Conversion in Virtex-II Pro Devices	Xilinx	Reference Design	V-IIP							
XAPP 651: SONET and OTN Scramblers/Descramblers	Xilinx	Reference Design	V-IIP							
XAPP 652: Word Alignment and SONET/SDH Framing	Xilinx	Reference Design	V-IIP							
XAPP660: Partial Reconfiguration of RocketIO Attributes using PPC405 core (DCR Bus)	Xilinx	Reference Design	V-IIP							
XAPP661: RocketIO Transceiver Bit-Error Rate Tester (BERT)	Xilinx	Reference Design	V-IIP							
XAPP662: Partial Reconfig. of RocketIO Attributes using PPC405 core (PLB or OPB bus) + RocketIO Transceiver Bit-Error Rate Tester (BERT)	Xilinx	Reference Design	V-IIP							
Basic Elements										
Binary Counter	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Binary Decoder	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Bit Bus Gate	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Bit Gate	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Bit Multiplexer	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
BUFE-based Multiplexer Slice	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
BUFT-based Multiplexer Slice	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Bus Gate	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Bus Multiplexer	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Comparator	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
FD-based Parallel Register	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
FD-based Shift Register	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Four-Input MUX	Xilinx	LogiCORE								S
LD-based Parallel Latch	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Parallel-to-Serial Converter	Xilinx	LogiCORE								S
RAM-based Shift Register	Xilinx	LogiCORE	V-IIP	V-II	V-E	V	S-3	S-IIE	S-II	
Register	Xilinx	LogiCORE								S
Three-Input MUX	Xilinx	LogiCORE								S
Two-Input MUX	Xilinx	LogiCORE								S
Prototype & Development Hardware Products										
CAN, LIN, and TTCAN Development Platform and Starter Kit (iDEV2)	Intelliga Integrated Design, Ltd.	AllianceCORE						S-IIE		
CPU + FPGA (Virtex/Spartan-II) MicroEngine Cards	Intrinsyc, Inc.	AllianceCORE								
CPU + FPGA (Virtex-II) MicroEngine Cards	Intrinsyc, Inc.	AllianceCORE		V-II						
JockoBoard SOC Virtex-II Prototyping Platform	RealFast Operating Systems AB	AllianceCORE		V-II						
LogiCRAFT Evaluation System	Xylon d.o.o.	AllianceCORE			V-E				S-II	
uPCI Reference and Development Platform	Intrinsyc, Inc.	AllianceCORE								
Virtex-II Pro PCI Platform FPGA Development Board	Amirix Systems, Inc.	AllianceCORE	V-IIP							
Virtex-II Pro Development Kit	Avnet Design Services	AllianceCORE	V-IIP							
XTENSA Microprocessor Emulation Kit (XT2000-X)	Tensilica, Inc.	AllianceCORE		V-II						

Visit the Xilinx IP Center for more details at *www.xilinx.com/ipcenter*

Avoid messy timing mistakes. Use Mentor Graphics® FPGA design tools.

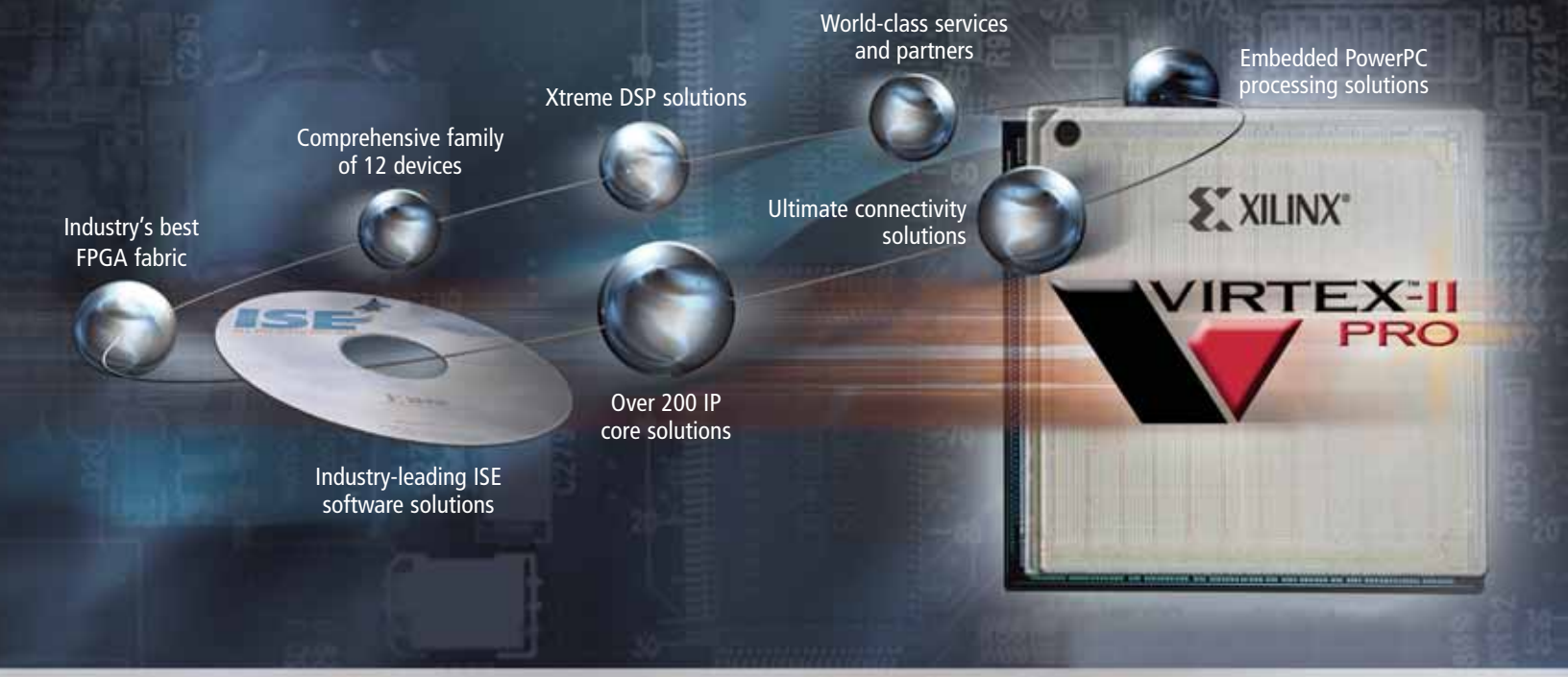
FPGA Design



PRECISION™ SYNTHESIS | As any FPGA designer can tell you, achieving precise timing is their single most critical challenge. That's why Mentor Graphics® created Precision™ Synthesis, a powerful new tool suite that lets designers close on timing faster than ever. Precision™ Synthesis provides the most comprehensive analysis for complex FPGAs with the only complete, built-in incremental timing analysis. Quickly find and optimize the most critical paths and avoid frustrating trial-and-error design iterations. Visit **www.mentor.com/fpga** today or call **877.387.5873** for more information on how Precision™ Synthesis can help you find the fastest path to a completed design.

Mentor Graphics®

LOWEST SYSTEM COST.
HIGHEST SYSTEM PERFORMANCE.



Virtex-II Pro™ Platform FPGAs deliver more capabilities and performance than any other FPGA.

In a single device, you get the highest density, most memory, best performance, and at no additional charge 400MHz PowerPC™ processors, plus 3.125 Gbps RocketIO™ or 10.3125 Gbps RocketIO X serial transceivers. This gives you the fastest DSP, connectivity and processing solutions in the industry.

The Power to Develop Your Design

The Xilinx ISE software tools are the easiest to use solution for high-density logic. With over 200 IP cores, ChipScope Pro debug environment, and compile times up to 6x faster than our nearest competitor, you can quickly take your Virtex-II Pro FPGA design from concept to reality.

Guaranteed 25% And Up To 80% Savings With Virtex-II Pro EasyPath Solutions

For high-volume system production, Xilinx offers a lower cost path with Virtex-II Pro EasyPath. You can immediately take advantage of dramatic cost reduction at no risk and no effort. Only Xilinx can offer you this flexibility for design and production.

VIRTEX-II PRO EASYPATH SOLUTIONS

Visit ***www.xilinx.com/virtex2pro*** today to get the price and performance you're looking for.

XILINX®
The Programmable Logic Company℠

www.xilinx.com/virtex2pro

FORTUNE® 2003
100 BEST COMPANIES TO WORK FOR

PN 0010764

Earth Friendly

Pb-free devices available now

Avoid Costly Re-Quals
Only PLD Company
Using 300mm
Wafers