WP416 (v1.0) April 24, 2012

Vivado Design Suite

By: Tom Feist

The Vivado™ Design Suite is a new IP and system-centric design environment that accelerates design productivity for the next decade of All-Programmable devices.

All-Programmable devices go beyond programmable logic and I/O, integrating various combinations of 3D stacked silicon interconnect technology, software programmable ARM® processing systems, programmable Analog Mixed Signal (AMS), and a significant amount of intellectual-property (IP) cores. These next generation devices enable designers to go beyond programmable logic to programmable systems integration, to incorporate more system functions into fewer parts, increase system performance, reduce system power, and lower BOM cost.

Built from the ground up over the last 4 years, the Vivado Design Suite attacks the key design bottlenecks in programmable systems integration and implementation to enable up to a 4X productivity advantage over competing development environments. For traditional logic-centric FPGAs, it also enables designs to be created more easily, meet timing more quickly, and automate—not dictate—the developer's preferred design flow.

Introduction

Vivado Design Suite has been under development since May of 2008, and in beta with over 100 customers since April 2011, leveraging technology from some key Xilinx acquisitions, including Hier Design, PwrLite Inc., and AutoESL Design Technologies, Inc. In addition, Vivado Design Suite introduces several *new* technologies, including a shared scalable data model, RTL synthesis, and place and route technology.

The core Vivado Design Suite technology is designed to scale to support massive devices with half the memory footprint and up to a 4X run-time advantage compared to competing programmable logic development environments.

Accelerated design integration is achieved through IP and system-centric design flows that quickly turn user designs and algorithms into reusable IP that can be assembled abstractly without errors.

Accelerated design implementation is achieved through analytical place and route technology in a tightly integrated environment that concurrently optimizes for congestion, wire length, and timing. Vivado Design Suite is an open design environment that embraces industry standards to leverage the designer's experience while broadening third-party support.

Accelerating Time to Integration

Today's FPGAS and EPP devices enable developers to implement complex systems on a single device. Xilinx 7 series FPGAs and Zynq EPPs come equipped with a wealth of dedicated hard IP blocks for implementing high-speed interfaces, memory interfaces, processing systems, signal processing, and video algorithms while being supported by extensive portfolios of soft IP from Xilinx, third parties, and customers themselves. Vivado Design Suite has been developed to simplify the task of integrating this IP to assemble, verify, and implement and system designs.

High-Level System Integration

The Vivado IP integrator is an interactive design and verification environment, enabling the user to build and verify a hierarchical system by graphically connecting IP provided by Xilinx, third parties, or the developer's propriety IP, using interface level connections, such as AXI, as illustrated in Figure 1. Support for hierarchical design entry matches typical designer flows, and eases creation and reuse of complex subsystems. Interactive design rule checking (DRC) checks guarantee that the system design is structurally correct by construction. In turn, the design itself can then be packaged as an IP and placed in the Vivado IP catalog for later reuse. The Vivado IP integrator supports processor or processor-less design creation equally well.

As a result of tight integration within the Vivado IDE, the Vivado IP integrator subsystems can be rapidly integrated into the overall design. Designers can also take advantage of the Vivado IDE's Tcl-based environment and build subsystems using scripts, or a combination of graphical and scripting methods. Advanced design services in the Vivado IP integrator perform DRC during design assembly, recognize and alert the designer to common design errors, handle automatic parameter propagation between IP blocks, and perform system-level optimizations.

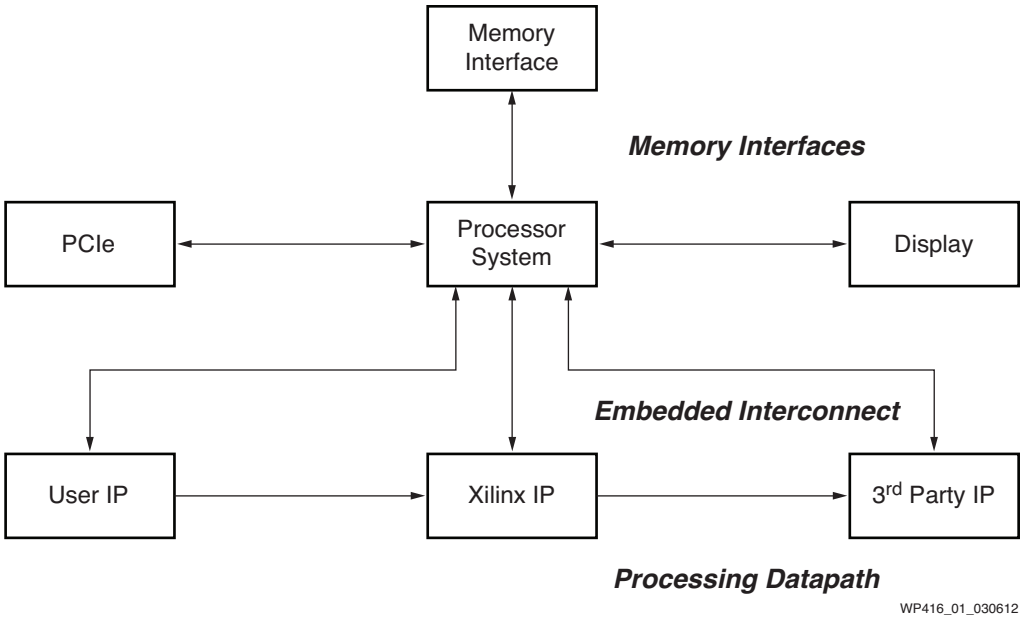

Figure 1: **Vivado IP Integrator**

The Vivado IP integrator accelerates IP and system-centric design implementation by providing the following:

- Tight integration within the Vivado IDE
  - Seamless inclusion of IPI subsystems into overall design
  - Rapid capture and packing of IPI designs for reuse
  - Tcl scripting as well as graphical design
  - Rapid simulation and cross-probing back between multiple design views
- Support for all design domains
  - Support for processor or processor-less designs
  - Integration of algorithmic and RTL-level IP
  - Combination of DSP, video, embedded, connectivity and logic
- Hierarchy support
  - Matches typical designer flows
  - Easy to reuse complex subsystems
- Advanced design services
  - DRCs on complex interface level connections during design assembly
  - Recognition and correction of common design errors
  - Automatic IP parameter propagation to interconnected IP
  - System-level optimizations

Package and Deployment of System-Level IP

The Vivado IP packager enables Xilinx, designers, and third-party IP providers to package a design with its constraints, test benches, and documentation to make it available in an extensible IP catalog on a local or shared drive.

Unique in the industry, Vivado Design Suite is the only FPGA/EPP design environment that provides the ability to package IP at any level of the design flow: RTL, netlist, placed netlist, even placed and routed netlist, to ensure performance. The Vivado Design Suite IP flow is illustrated in Figure 2.

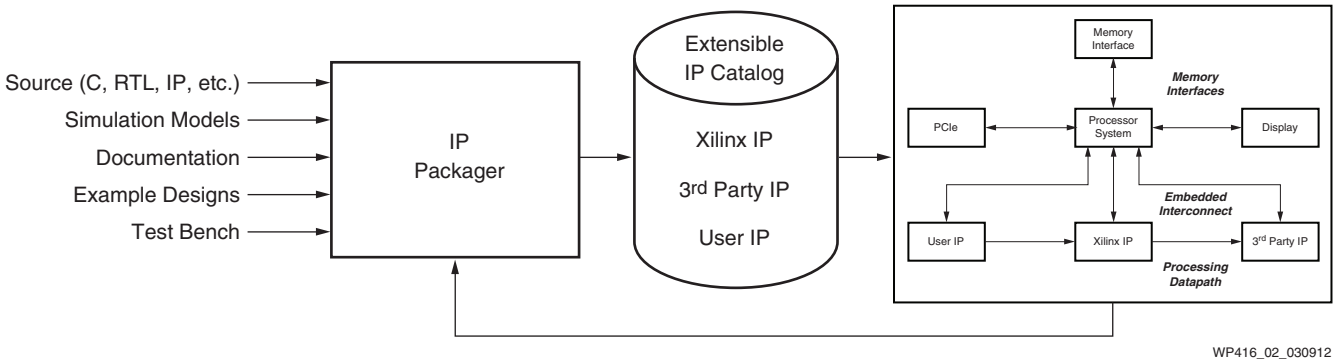

Figure 2: **Vivado Design Suite IP Flow**

The Vivado IP catalog provides an inventory of Xilinx, third-party, and intra-company IP that can be shared across a design team, a division, or a company in an easy-to-use manner. Based on the IP-XACT standard format, all Xilinx IP — including embedded, DSP, interface, building block, and connectivity IP — is consolidated into a view that can be accessed consistently, regardless of the end application being developed. Support for multiple physical locations, including shared networked drives, allows users or organizations to leverage a consistent IP deployment environment for third-party or internally generated IP.

Vivado Design Suite shatters the RTL design productivity plateau by providing more effective design reuse. The Vivado IP catalog is an extensible IP repository, providing the framework for an IP and system-centric design flow. This catalog consolidates IP from all sources, including all Xilinx IP, IP obtained from third parties, and end-user designs targeted for reuse, into a single environment that can be deployed throughout the user's company with a consistency that simplifies its use. Support for open industry IP standards enables third-party vendors to expand their IP portfolios, which can then be included in the Vivado IP catalog. Users can package their own C/C++/SystemC or MATLAB®/Simulink® algorithms or system designs into the IP catalog using Vivado High-Level Synthesis (HLS) or System Generator for DSP with the Vivado IP packager. Once cataloged, the integration of this broad portfolio of IP into verified, system-level designs can then be accomplished abstractly at the interface level using Vivado IP integrator with design rule checking that minimizes user errors.

Integrated Verification of Mixed-Language IP

Mixed-language designs assembled from IP can be verified using the Vivado simulator. This simulator is tightly integrated into the Vivado Integrated Design Environment (IDE) to shorten learning curves. Users can launch behavioral simulations with a single click of the mouse and view results using a common waveform viewer. Simulations are accelerated at the behavioral level, using a new performance-optimized simulation kernel that executes up to 3X faster than the Xilinx® ISE® Design Suite simulator. Gate-level simulations can also be accelerated up to 100X using hardware co-simulation.

Accelerating IP Development with High-Level Synthesis

Advanced algorithms used in communications, medical, defense, high-performance computing, and consumer applications are more sophisticated than ever before. This complexity, combined with aggressive development and verification schedules, challenges even the most experienced RTL teams. Vivado HLS (built on AutoESL™ tool technology) accelerates design implementation and verification by enabling C, C++, and System C specifications to be directly synthesized into VHDL or Verilog RTL, after exploring a multitude of micro-architectures based on design requirements. Functional simulation can be performed in C, providing an order of magnitude acceleration over VHDL or Verilog simulation. This provides designers and system architects with a faster and more robust way of delivering quality designs. In one example, a video motion estimation algorithm, the C input to Vivado HLS executed 10 frames of video data in 10 seconds, while the corresponding RTL model took roughly two days to process the same 10 video frames.

The Vivado HLS flow is illustrated in Figure 3.

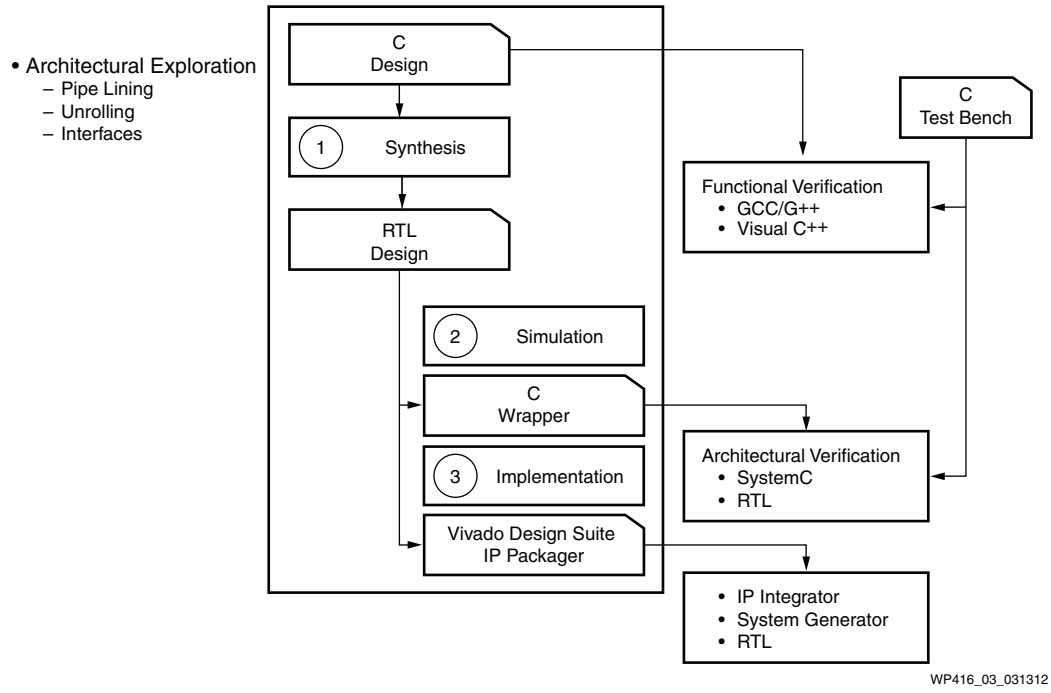

Figure 3: **Vivado HLS Flow**

The keys benefits of Vivado HLS are:

- Integrated Eclipse C Development Tool (CDT) environment to specify, compile, simulate, and debug C/C++/SystemC:
  - C with arbitrary precision specification and simulation
  - C++ with classes for complex data types and interface models
  - SystemC for explicit parallelism and multiple clock domains
- Algorithm verification speeds up to 100X faster than RTL
- Quality of results that rivals hand-coded RTL:
  - Automatic extraction of parallelism at instruction level and task level
  - Flexible implementations of floating point algorithms
  - Architecture-aware inference of DSP48, on-chip memories, and SRL infrastructures
  - Fast design exploration that creates results in VHDL, Verilog, and SystemC
  - Micro-architecture exploration that enables easy trade-off of utilization, power, throughput, and latency
- Automated RTL verification through co-simulation with original C-based test bench
- Plug-and-Play IP generation streamlining integration:
  - AXI4-based interface synthesis
  - Comprehensive packaging of RTL results with the Vivado IP packager

For an industry-independent report on the value of high-level synthesis, see the BDTI white paper: *An Independent Evaluation of High-Level Synthesis Tools for Xilinx FPGAs.* [Ref 2]

Standards-Based IP and System-Centric Environment

Xilinx has taken advantage of industry standards when developing Vivado Design Suite to improve ease of use by supporting already familiar standards for constraints and interfaces as well as to expand third-party ecosystem support for IP and tool flows.

Vivado Design Suite supported standards include:

- **AMBA® AXI4 interconnect (AXI):** ARM Standard for defining IP interconnect.
- **Liberty Modeling Library (.lib):** Used by virtually all EDA implementation, analysis, and library characterization tools as the library model exchange for timing, noise, power, and test behavior.
- **Synopsys Design Constraints (SDC):** Widely-used design constraints format that describes the "design intent" and surrounding constraints for synthesis, clocking, timing, power, test, and environmental and operating conditions. SDC has been in use and evolving for more than 20 years, making it the most popular and proven format for describing design constraints.
- **IP-XACT:** An XML format that defines and describes electronic components and their designs. IP-XACT was created by the SPIRIT Consortium as a standard to enable automated configuration and integration through tools—approved as IEEE Std 1685-2009. [Ref 1]
- **Tool Command Language (Tcl):** A powerful interpreted programming language. Commonly used for rapid prototyping of scripted applications.

- **Verilog Netlists:** Provided to convey connectivity information and provide instances, nets, and some attributes for interoperability between EDA tools.
- **IEEE P1735:** IP Security standard that specifies embeddable and encapsulating markup syntaxes for design IP encryption and rights management.

Accelerating Time to Implementation

Vivado Design Suite accelerates the implementation process by delivering more turns per day while helping to reduce the number of design iterations needed. A revolutionary, shared, scalable data model delivers up to 4X faster run times and half the memory footprint of competing solutions. Vivado Design Suite provides early access to critical design analysis metrics such as power, timing, and resource utilization with increasing accuracy as the design progresses through the elaboration, synthesis, and placement and routing processes. These metrics allow design and tool setting modifications to occur earlier in the design processes where they have less overall schedule impact. Vivado Design Suite's analytical place and route technology delivers more predictable design closure by concurrently optimizing for congestion, wire length, and timing. Vivado Design Suite's incremental design flow lets users quickly make changes late in the design cycle while preserving the performance of the design.

Vivado Design Suite Shared Scalable Data Model

Vivado Design Suite employs a new shared, scalable data model to accommodate the largest programmable logic devices. This single data model is used throughout the implementation process, enabling users to gain visibility into key design metrics such as timing, power, resource utilization, and routing congestion much earlier in the design process. These estimates become progressively more accurate throughout the design flow, enabling faster design closure with fewer iterations.

The data model is illustrated in Figure 4.

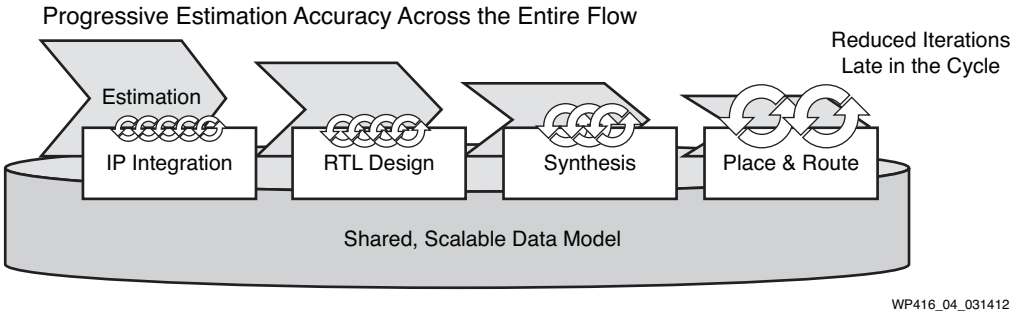

Figure 4: **Vivado Design Suite Shared Scalable Data Model**

Analytical Place and Route Engine

Traditional FPGA implementation tools use simulated annealing as a placement optimization algorithm. This algorithm starts with a random placement and then tries to optimize a cost function — typically a timing cost — by exploring part of the solution space with *local* placement moves and calculating the cost after each move. This is illustrated in Figure 5.

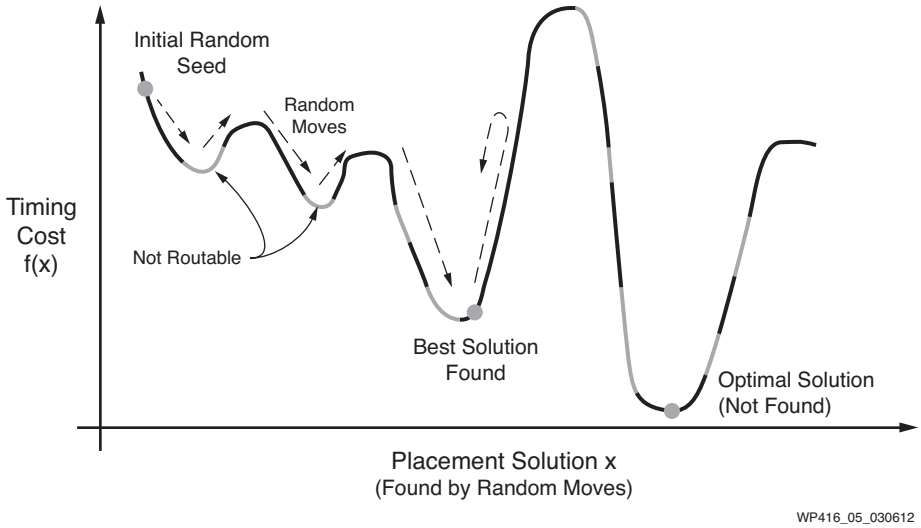

Figure 5: **Traditional Simulated Annealing Algorithm**

Given the random nature of the initial solution and the subsequent moves, traditional simulated annealing algorithms do not scale to million-LUT designs. Because it works with local moves, this traditional optimization method is blind to global design metrics, such as the level of congestion of the solution or the total wire length.

Vivado Design Suite models the placement solution space into a large mathematical equation and uses an analytical solver to find a solution that minimizes a given cost function. An optimal placement solution depends on multiple dimensions, such as timing (T), wire length (W), and congestion metrics (C). Vivado Design Suite uses a multi-variable cost function to find the optimal placement, allowing the designer to quickly find a routable solution that maximizes performance while minimizing wire length, and therefore, dynamic power. This is illustrated in Figure 6.

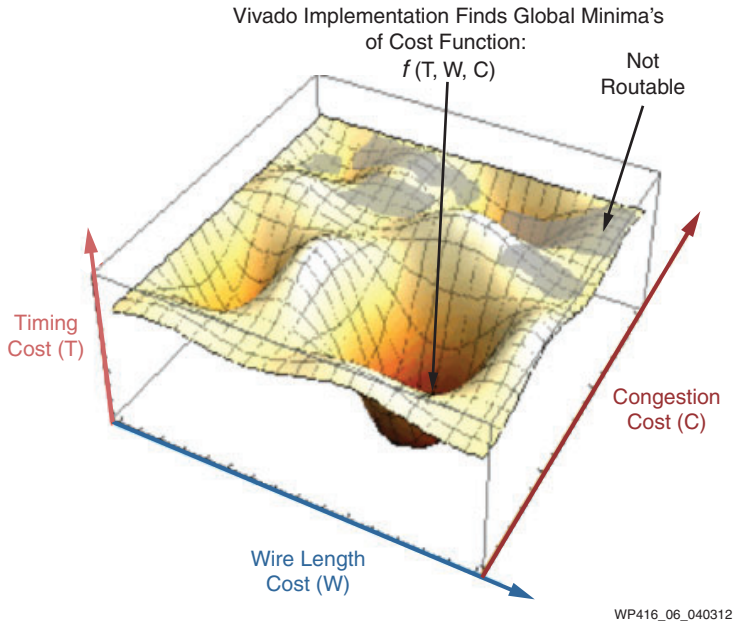

Figure 6: **Vivado Design Suite Analytic Place and Route**

Table 1 compares the traditional place and route processes to the Vivado placer.

Table 1: **Traditional Place and Route Compared to Vivado Placer**

	Traditional Place and Route	Vivado Place and Route
"Cost" Criteria	1 Dimension: Timing minimization	3 Dimensions: Timing, congestion, wire length minimization
Primary Algorithm	Simulated Annealing: Local, iterative search based on initial random seed	Analytical: Solves an equation to globally minimize all dimensions
Run Time	Unpredictable (due to random nature of algorithm): Increases exponentially with congestion	Very predictable: Grows linearly with design size
Scalability	Poor results as design approaches 1M logic cells	Handles 10M+ logic cells with predictable results

The graph shown in Figure 7 highlights both the run-time advantage and the predictable behavior of the Vivado Design Suite place and route engine. Run times are consistently up to 4X faster than alternative solutions, while the variance in results is much tighter, enabling design closure with fewer iterations.

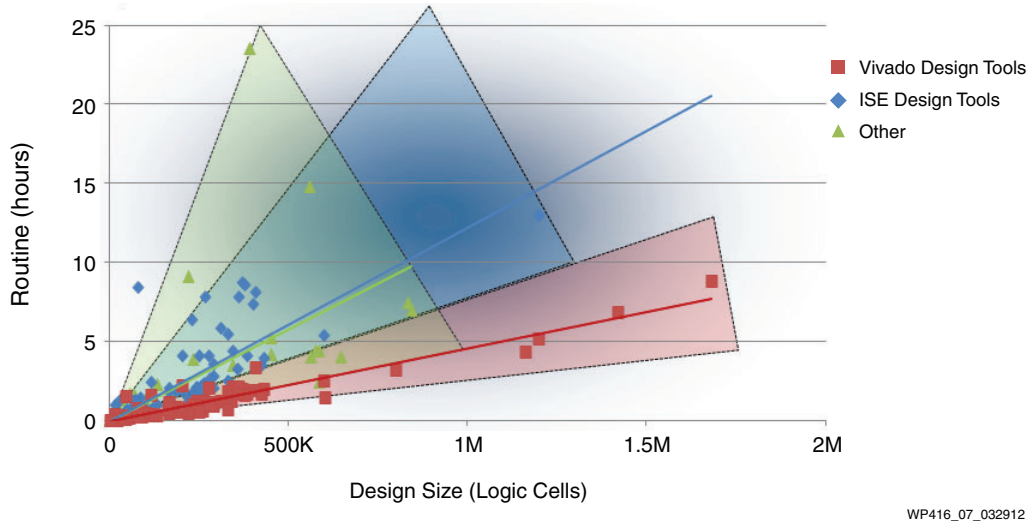

Figure 7: **Predictable Place and Route Results**

Hierarchical Design Flows

Vivado Design Suite also supports hierarchical design flows that enable team design, design reuse, and partial reconfiguration.

In a team design, blocks can be implemented independently and simultaneously. This allows each block to be individually designed, implemented, and verified out of context with the rest of the design. When all blocks are complete, the entire design is brought together in an assembly run with the top-level design, while maintaining the results for each module.

As illustrated in Figure 8, this "out of context" flow enables placed or routed blocks to be imported once or multiple times in any designs. This facilitates design reuse and the ability to create system-level IP, enhancing productivity for anyone targeting FPGAs or EPPs.

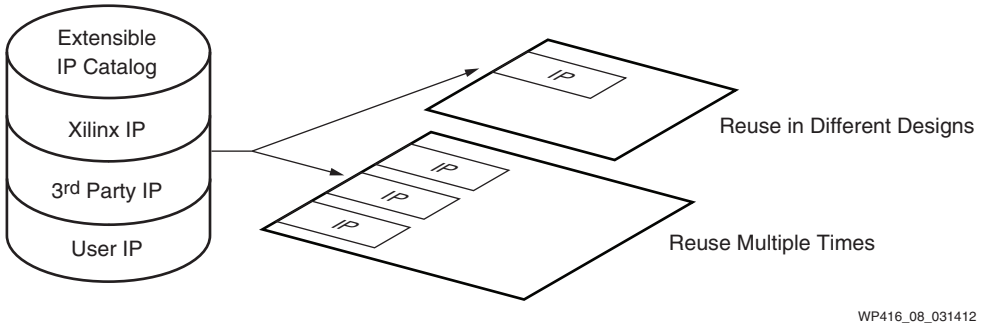

Figure 8: **IP Reuse**

These reusable blocks maintain their timing and do not consume additional CPU cycles during implementation.

Partial Reconfiguration

Xilinx partial reconfiguration extends the inherent flexibility of the FPGA by allowing specific regions of the FPGA to be reprogrammed with new functionality while applications continue to run in the remainder of the device. Partial reconfiguration addresses three fundamental needs by enabling the designer to:

- Reduce cost and/or board space
- Change a design in the field
- Reduce power consumption

Vivado Design Suite extends the software support established in prior generations of Xilinx tools, adding significant enhancements to the quality and capabilities of this technology.

Faster Design Closure with Design Analysis

The key to providing more turns per day is Vivado Design Suite's early estimates for power, performance, and utilization. At each stage of the design (post synthesis, post placement, and post routing), estimates can be generated to gauge how well the design requirements are being met. The further along in the flow, the more accurate the estimation.

Vivado Design Suite also supports “what-if analysis,” which is the ability to try various configurations of the design and compare reports at the same stage. For example, a designer can easily generate multiple reports to estimate power under different operating conditions or activity patterns and compare results side by side. The designer can also add timing exceptions and re-run timing analysis without reimplementing the design.

Easier Design Debug with Cross-Probing

Vivado Design Suite is designed and structured to work off a common data model in memory. This architecture enables extensive cross-probing between design sources, schematic views, hierarchy browsers, design reports, messages, floor plan, and Vivado Device Editor views. This unique capability enables faster debug and timing closure by providing the design feedback, pointing to where in the design the issues are coming from. This is illustrated in Figure 9.

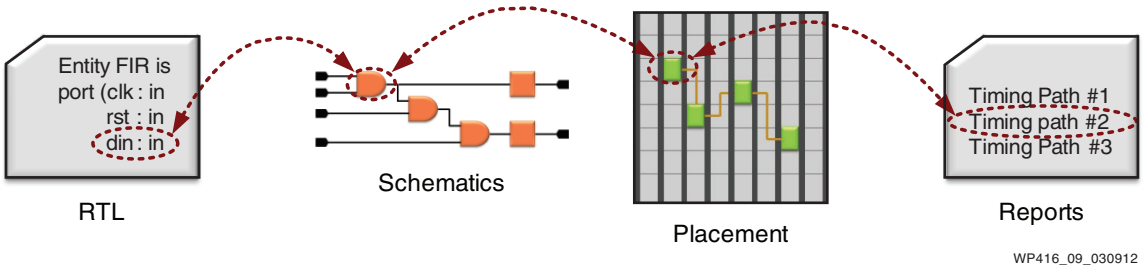

Figure 9: **Vivado Design Suite Design Debug**

Incremental Implementation with Vivado Device Editor

Using the Vivado device editor on a routed design, a designer has the power to make incremental changes, such as moving instances, rerouting nets, tapping a register to a primary output for debug with scope, or changing the parameters on a Digital Clock Manager (DCM), late in the design cycle without needing to go back through synthesis and implementation. No other FPGA design environment offers this level of flexibility. Once made, these edits can be saved to Tcl commands and included in repeatable implementation scripts.

Incremental flows allow for small changes to be quickly processed by reimplementing only a small part of the design, making iterations faster after each change. They also enable performance preservation after each incremental change, thus, reducing the need for multiple design iterations.

Design Flows Tailored to the Designer's Needs

A key consideration when designing Vivado Design Suite was to make it intuitive for all users, from new to expert. This includes the individual designer who works with an Artix®-7 device to handle a few thousand LUTs to large design teams using Virtex®-7 2000T FPGAs, handling over a million LUTs. Ease of use is addressed in a variety of ways, including:

- Preconfigured pushbutton flows for new users
- Advanced analysis-driven flows for expert users
- Support for already familiar industry standards
- Exceptional cross-probing and design navigation from RTL to layout to reports
- Highly readable and interactive schematic viewer that highlights RTL coding effects
- Simplified method for design creation and system assembly

To accommodate batch mode users, every action in the GUI issues an equivalent Tcl command, so that anything that can be done in the GUI can also be automated by scripting.

For new users, an intuitive toolbar guides the designers through all the setup and implementation steps necessary to generate a programming file from RTL source. Users who want a pushbutton flow and do not need access to more advanced tool options or design analysis can choose to proceed directly to the final implementation process step, and in doing so, perform all implementation steps necessary to generate the programming file.

Figure 10 illustrates three use models:

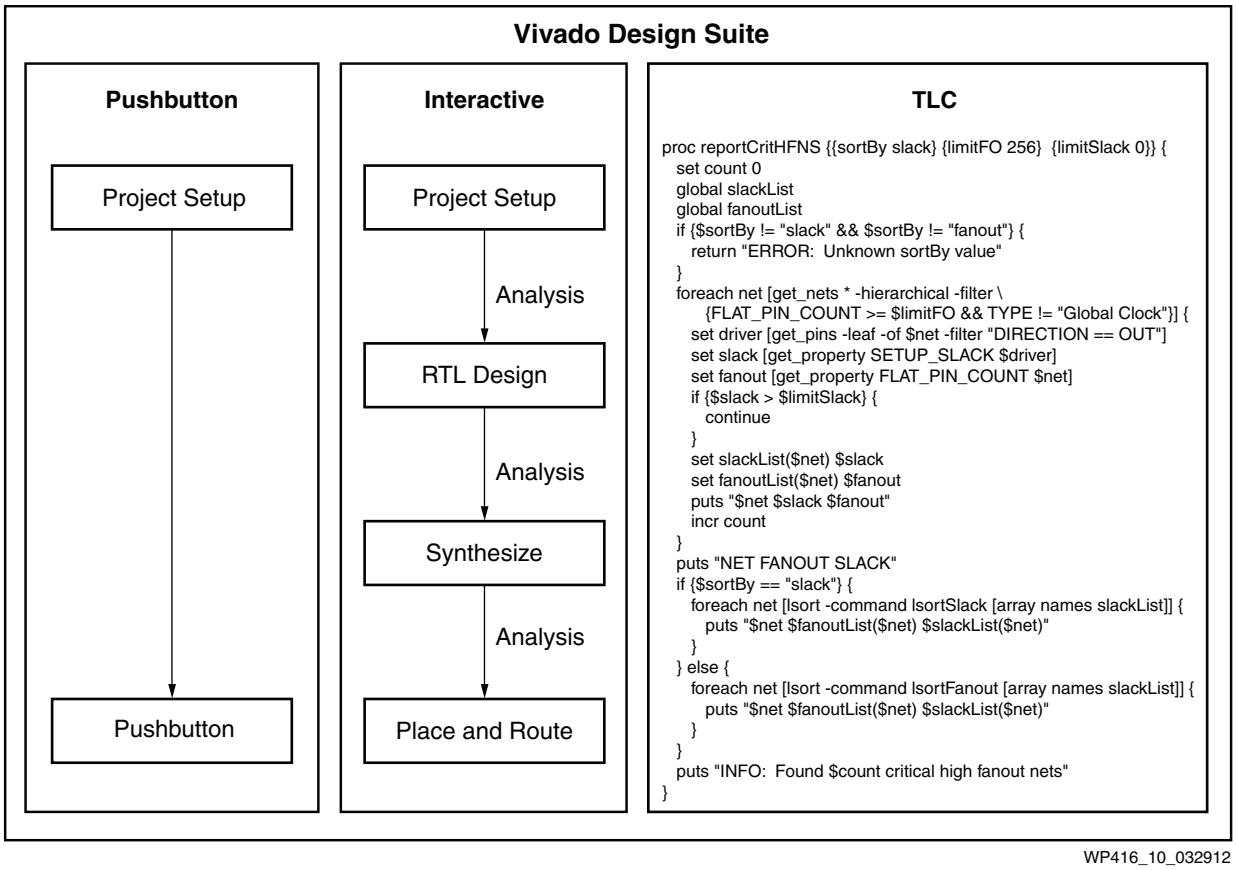

Figure 10: **Vivado Design Suite Use Models**

Alternatively, expert users can progress step-by-step through the implementation process using a common data model that provides early access to key design information — such as resource utilization, timing, power, and noise — that becomes increasingly accurate as the design progresses through the flow. This analysis-driven design flow helps to identify design issues early in the implementation flow where they can be addressed with shorter iteration times, ultimately reducing the number of design iterations required to complete a design.

In addition, Vivado Design Suite allows designers to configure, launch, and monitor multiple synthesis and implementation runs either locally or on remote Linux servers. This enables faster "What If" analysis with different command options, constraints, and devices.

Vivado Design Suite Software Editions

Vivado Design Suite is offered in three variants, as shown in Table 2:

Table 2: **Vivado Software Offerings**

Feature	WebPACK™ Tool (Device-Limited Free Edition)	Design	System
Vivado IDE	✓	✓	✓
Vivado IP Catalog	✓	✓	✓
Vivado Synthesis	✓	✓	✓
Power Optimizer	✓		
Vivado Simulator	✓	✓	✓
Vivado Implementation	✓	✓	✓
Vivado RTL Analysis Vivado Synthesized Design Analysis Implemented Design Analysis	✓	✓	✓
Vivado Static Timing Analysis	✓	✓	✓
Vivado Pin Planner	✓	✓	✓
Vivado Device Editor	✓	✓	✓
Vivado IP Packager	✓	✓	✓
Vivado IP Integrator	✓	✓	✓
Software Development Kit (SDK)	✓	✓	✓
Logic Analyzer Serial I/O Analyzer	✓	✓	✓
Vivado High-Level Synthesis			✓
System Generator for DSP			✓

Additional Information

To learn more about the Vivado Design Suite, go to www.xilinx.com/vivado or talk to a local Xilinx representative.

References

1. IEEE Standards Association: IEEE Standard 1685-2009, *IEEE Standard for IP-XACT, Standard Structure for Packaging, Integrating, and Reusing IP within Tool Flows* http://standards.ieee.org/findstds/standard/1685-2009.html
2. Berkeley Design Technology, Inc.: *An Independent Evaluation of High-Level Synthesis Tools for Xilinx FPGAs* http://www.xilinx.com/technology/dsp/BDTI_techpaper.pdf

Revision History

The following table shows the revision history for this document:

Date	Version	Description of Revisions
04/24/12	1.0	Initial Xilinx release.

Notice of Disclaimer