

# Accelerating Architecture Exploration for FPGA Selection and System Design

You can create an optimized specification that achieves performance, reliability, and cost goals for use in production systems.

by Deepak Shankar  
 President and CEO  
 Mirabilis Design Inc.  
[dshankar@mirabilisdesign.com](mailto:dshankar@mirabilisdesign.com)

Performance analysis and early architecture exploration ensures that you will select the right FPGA platform and achieve optimal partitioning of the application onto the fabric and software. This early exploration is referred to as rapid visual prototyping. Mirabilis Design's VisualSim software simulates the FPGA and board using models that are developed quickly using pre-built, parameterized modeling libraries in a graphical environment.

These library models resemble the elements available on Xilinx® FPGAs, including PowerPC™, MicroBlaze™, and PicoBlaze™ processors; CoreConnect; DMA; interrupt controllers; DDR; block RAM; LUTs; DSP48E; logic operators; and fabric devices. The components are connected to describe a given Xilinx Virtex platform and simulated for different operating conditions such as traffic, user activity, and operating environment.

More than 200 standard analysis outputs include latency, utilization, throughput, hit ratio, state activity, context

switching, power consumption, and processor stalls. VisualSim accelerates architecture exploration by reducing typical model development time from months to days.

I can illustrate the advantages of early architecture exploration with an example from one of our customers, who was experiencing difficulty with a streaming media processor implemented using a Virtex™-4 device. The design could not achieve the required performance and was dropping every third frame. Utilization at all of the individual devices was below 50%. A visual simulation that combined both the peripheral and the FPGA identified that the video frames were being transferred at the same clock sync as the audio frames along a shared internal bus.

As the project was in the final stages of development, making architecture changes to address the problem would have delayed shipment by an additional six months. Further refinement of the VisualSim model found that by giving the audio frames a higher priority, the design could achieve the desired performance, as the audio frames would also be available for processing. The project schedule was delayed by approximately 1.5 months.

If the architecture had been modeled early in the design cycle, the design cycle could have been reduced by 3 months, eliminating the 1.5-month re-spin to get to market approximately 5 months sooner. Moreover, with a utilization of 50%, control processing could have been moved to the same FPGA. This modification might have saved one external processor, a DDR controller, and one less memory board.

## Rapid Visual Prototyping

Rapid visual prototyping can help you make better partitioning decisions. Evaluations with performance and architectural models can help eliminate clearly inferior choices, point out major problem areas, and evaluate hardware/software trade-offs. Simulation is cheaper and faster than building hardware prototypes and can also help with software development, debugging, testing, documentation, and maintenance. Furthermore, early partnership with customers using visual prototypes improves feedback on design decisions, reducing time to market and increasing the likelihood of product success (Figure 1).

A design-level specification captures a new or incremental approach to improve system throughput, power, latency, utiliza-



PowerPC will be sharing the bus with the MicroBlaze processor, the effective bus throughput is a function of the number of data requests and the size of the local block RAM buffers. For example, you could enhance the MicroBlaze processor with a co-processor to do encryption at the bit level in the data path. You could also use the CoreConnect bus to connect the peripheral SDRAM to the PowerPC while the DDR2 is used for the MicroBlaze processor.

You can reuse the VisualSim architecture model for exploration of the software design, identifying high-resource consumption threads, balancing load across multiple MicroBlaze processors, and splitting operations into smaller threads. If a new software task or thread has data-dependent priorities, exploration of the priorities and task-arrival time on the overall processing is a primary modeling question. If you change the priority on a critical task, will this be sufficient to improve throughput and reduce task latency? In most cases, this will be true, but there may be a relative time aspect to a critical task that can reduce latencies on lower priority tasks such that both benefit from the new ordering. If peak processing is above 80% for a system processing element, then the system may be vulnerable to last-minute tasks added, or to future growth of the system itself.

### Model Construction

System modeling of the Layer 3 switch (Figure 3) starts by compiling the list of functions (independent of implementation), expected processing time, resource consumption, and system performance metrics. The next step is to capture a flow diagram in VisualSim using a graphical block diagram editor (Figure 3). The flow diagrams are UML diagrams annotated with timing information. The functions in the flow are represented as delays; timed queues represent contention; and algorithms handle the data movement. The flow diagram comprises data processing, control, and any dependencies.

Data flow includes flow and traffic management, encryption, compression, routing, proxy rules, and TCP protocol

handling. The control path contains the controller algorithm, branch decision trees, and weighted polling policies. VisualSim builds scenarios to simulate the model and generate statistics. The scenarios are multiple concurrent data flows such as connection establishment (slow

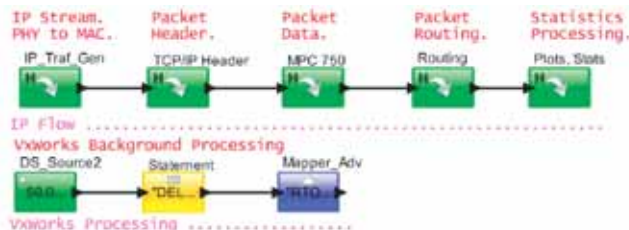


Figure 3 – Flow chart describing the application flow diagram in VisualSim

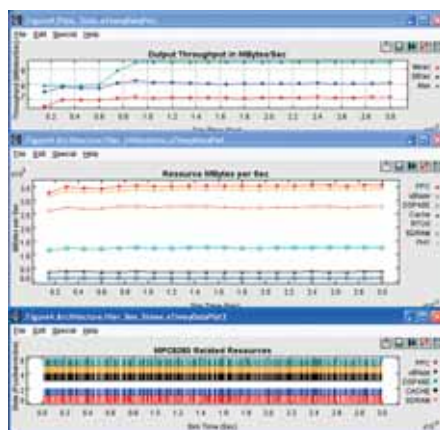


Figure 4 – Analysis output for the Layer 3 switch design

path); in-line data transfer after setup of secure channel (fast path); and protocol- and data-specific operation sequences based on data type identification.

You can use this model of the timed flow diagram for functional correctness and validation of the flows. VisualSim uses random traffic sequences to trigger the model. The traffic sequences are defined data structures in VisualSim; a traffic generator emulates application-specific traffic. This timed flow diagram selects the FPGA platform and conducts the initial hardware and software partitioning. The flow diagram model defines the FPGA components and peripheral hardware using the FPGA Modeling Toolkit.

The functions of the flow diagram are mapped to these architecture components. For each function, VisualSim automatically collects the end-to-end delay and number of packets processed in a time period. For the architecture, VisualSim plots the average processing time, utilization, and effective throughput (Figure 4). These metrics are matched against the requirements. Exploration of the mapping and architecture is possible by varying the link and replacing the selected FPGA with other FPGAs.

The outcome of this effort will be selecting the right FPGA family, correctly sizing the peripherals and the right number of block RAMs, DSP blocks, and MicroBlaze processors. You can add overhead to the models to capture growth requirements and ensure adequate performance.

### Conclusion

Early architecture exploration ensures a highly optimized product for quality, reliability, performance, and cost. This provides direction for implementation plans, reduces the amount of tests you need to conduct, and has the ability to shrink the development cycle by almost 30%.

VisualSim libraries of standard FPGA components, flow charts defining the behavior, traffic models, and pre-built analysis probes ensure that system design is no longer time consuming, difficult to perform, and providing questionable results. The reduction in system modeling time and availability of standard component models provides a single environment for designers to explore both hardware and software architectures.

For a free 21-day trial of the FPGA Modeling Toolkit, including the MicroBlaze and PowerPC models, register at [www.mirabilisdesign.com/webpages/evaluation/mdi\\_evaluation.htm](http://www.mirabilisdesign.com/webpages/evaluation/mdi_evaluation.htm). To learn more about VisualSim, visit [www.mirabilisdesign.com](http://www.mirabilisdesign.com), where there are models embedded in the HTML pages. You can modify parameters and execute from within your web browser without having to download custom software.