Consumer, mobile, networking and storage systems with multi-core processors are rapidly becoming more complex, making architecture decisions increasingly critical, and impacting the design’s competitive advantage. Configuring and verifying multi-core HW/SW architectures, and ensuring that the system can carry its load and data traffic capacities, are all critical tasks.

Designers can now use SystemC transaction level modeling (TLM) methodology to model their entire system, validate its functionality, quickly analyze various architectural tradeoffs among power, performance and area, and create virtual platforms for SW development and HW/SW integration.

Key questions to be answered at the architectural level include: Can the architecture deliver the necessary functionality and meet user expectations? Can the system meet performance and power consumption goals? Can the system be effectively implemented? Can software run correctly and efficiently on the target architecture?

Vista Architect is a complete TLM 2.0-based solution for architecture design, analysis and verification enabling system architects and SoC designers to make viable architecture decisions. This is accomplished by prototyping and analyzing complex systems to ensure optimized architectures, shorter implementation time and first pass success.

Low Power

These days everyone is concerned about power consumption. In concept, while you would like to reduce power consumption as much as possible, at the end of the day it is about balancing the low power requirement against meeting the system functionality, performance, and manufacturability.

Vista ESL breakthrough solution lets you tackle the power requirements early at the architectural level and allows designers to optimize power, performance and functionally way before committing to implementation.

Integrate with Software

Validation of hardware dependent software early in the process is a major objective of software development teams. With Vista Architect, users can test and debug the hardware driven by software or produce a virtual platform to run firmware, operating systems or hardware dependent software applications.

Link with OVM RTL Verification

While ESL and RTL may use different languages (such as SystemC and SystemVerilog respectively), and serve different use cases, the ability to link and reuse elements from both domains is important and offers a broader and more complete verification solution. Transaction level models created at the system level can seamlessly drive RTL sub-systems or be used as the executable specification (reference model) against which the RTL can be automatically verified. OVM defines such a methodology and flow that effectively reuses the transaction level models created for ESL design at the OVM RTL verification stage.
Scalable Transaction Level Modeling Methodology

Transaction level modeling (TLM) provides an abstract design methodology that supports modeling, validation, analysis and implementation. Mentor Graphics is offering a TLM 2.0 Scalable Modeling Methodology, based on a layered approach that separates communication, functionality and power/timing. The layered approach allows a model to maintain a single functional description throughout the entire ESL design cycle all the way to implementation.

The model contains a functional un-timed layer that defines the model behavior for “what” it does, in addition to the function the timing/power layer reflects “how” the function is implemented. The “timing” layer essentially captures the timing associated with a given function’s micro-architecture implementation and reflects the latency of computation, the pipelining across ports or the function’s response time (such as number of wait states).

During the design process, timing and power accuracy evolve and the model is refined from an abstract un-timed representation into a detailed implementation view of the target micro-architecture, all of which is represented within a single model.

Using this layered approach also allows the same model to be switchable during simulation, targeting fast loosely-timed software execution (in “LT” mode) or more detailed approximately-timed simulation for hardware verification, performance and power analysis (in “AT” timed mode).

Benefits

- Innovative TLM 2.0 modeling approach
- Layered modeling methodology
- Separation of communication, function and timing/power layers
- Enables incremental model refinement for timing and power
- Compatible with hardware verification and software execution requirements
- Consistent from system level to implementation

Create Transaction Level Models

Vista Model Builder, a sub-set of the Vista Architect solution, facilitates transaction level model creation, allowing users to efficiently create complex models using intuitive mechanisms and a set of pre-defined modeling base classes. Vista Model Builder is augmented with a new scalable modeling methodology based on TLM 2.0, where communication, functionality, and timing/power attributes are independently modeled. This important modeling practice allows a single functional model to be maintained throughout the design cycle at various implementation phases and through alternative design options. Vista Model Builder automates the process of modeling functionality with a set of TLM classes and a convenience layer for more efficient and guided behavioral modeling. A TLM code skeleton is automatically derived from a set of ports, registers and memory declarations generating compact SystemC source code compliant with TLM 2.0, so users can completely model the internal behavior.

Timing and power can be specified in a top-down manner through a set of powerful policies. The timing policies are used to intuitively model the timing of a desired micro-architecture including latencies, pipelining and wait states. Power policies can be defined for static leakage power, clock tree power, and dynamic power per transaction type. Using timing and power policies at the transaction level produces a transaction level timing and power model that is reactive to incoming traffic and inner states. The transaction level model will have the correct timing/power behavior in the context of system level simulation.

This approach allows users to quickly explore various complex micro-architecture alternatives in the system context with minimal coding effort while keeping the code representing the functionality intact.

Benefits

- Automated TLM 2.0 model generation
- Reduces modeling effort
- Separates functionality from communication
• Classes for behavioral modeling
• Timing/Power powerful policies
• Eases the exploration of various micro-architectures

Assemble and Configure the System

During the architecture design phase, models can be intuitively instantiated and assembled into a transaction level platform that represents various architecture configurations including interconnect layering and memory hierarchies. Vista’s powerful block diagram editor provides intuitive graphical platform assembly, editing and visualization. By understanding the TLM 2.0 connectivity semantics, Vista is capable of automatically generating the SystemC code representing the connectivity among all TLMs. Vista produces a simulation model that represents the assembled transaction level platform.

Benefits
• Intuitive graphical platform assembly editing and visualization
• Automatic creation of transaction level platform executable code

Verify and Debug the System

Vista Architect offers the industry’s most advanced SystemC debug toolset designed to validate and debug SystemC transaction level platforms. At the architecture level, verification is focused on validating the correct interaction among various IP and appropriate flow of data.

Vista Architect presents innovative debugging and tracing concept that focuses on high-level system debug and data flow analysis. It helps users understand how data is processed and passed through the system and its resources, understand the sequence of events, flow control and process scheduling.

Unique debugging mechanisms tailored for TLM 2.0 and SystemC/C/C++ modeling allow users to trace transactions, sequence of events and process execution within a familiar hardware debugging platform. Users can view design hierarchies as well as class hierarchies and understand how C/C++ data objects and functions are assigned and executed over time or even within delta cycles. All debug and analysis capabilities are supported without any source code instrumentation and naturally link with any existing SystemC design flow.

Benefits
• Transaction-level viewer for TLM debug and data tracing
• Understand data and control flow in complex systems

• View design and class hierarchies
• Unique tracking of process activity during run time
• Waveform traces C/C++ data objects with delta cycle resolution
• Comprehensive event sequence debugging
• SystemC debug switches between hardware and C/C++ views

Analyze and Optimize

Powerful Analysis Toolset

Vista Architect offers a powerful analysis and reporting toolset that allows users to intuitively analyze different performance and power metrics. Vista’s timing/performance analysis tools allow users to look at load peaks, average latencies, throughput and utilization on any port, bus or sub-system without any manual instrumentation. The comprehensive power analysis tools report dynamic, static and clock power for the entire system, instance power, mean and peak power profile diagrams.

Vista Architect enables users to rapidly analyze the system power consumption and performance under different system level scenarios and traffic loads. The scalable modeling approach supported in Vista Architect enables design teams to manage timing and power budgets from concept down to the desired implementation. This approach ensures that the silicon area is optimized and that the system is able to carry the data capacities for a given application. It also ensures that the system architecture is scalable to support future derivations of the product.

Exercise Various Scenarios

Complex data packets can be easily created and tagged with an “ID”, and then traced and analyzed as they propagate through the system. This unique capability allows users to clearly understand the flow of data through the system and the impact of various loading scenarios.

Users can exercise statistical and randomized data traffic or run SW-driven traffic, testing realistic use case scenarios. This allows for detailed analysis of the hardware and software domains tradeoffs and optimizing the hardware/software boundaries in the architecture.

With Vista Architect users can apply the scalable transaction level modeling approach for improved modeling and simulation efficiencies. They can dynamically switch from a pure loosely-timed (“LT”) simulation to approximately-timed (“AT”) simulation.
Quickly Change the Micro-architecture

The unique layered approach for modeling timing and power enables users to quickly change the timing policies for each micro-architecture model and test various configurations and pipeline strategies while keeping the functionality intact. Users can refine the timing and power accuracy based on the target bus protocol from high-level approximation down to precise timing in a matter of minutes.

Benefits

- Set of configurable TLM 2.0-based architecture blocks
- Advanced analysis and visualization

More information is available on our website www.mentor.com/vista