

# Cadence Virtual System Platform

## Virtual Prototyping Solution

Part of the Cadence® System Development Suite, the Cadence Virtual System Platform enables pre-RTL software development, functional verification, and system analysis and optimization before committing to hardware micro-architecture. It automates the process of creating a virtual prototype, debugging software with the virtual prototype, and deploying the virtual prototype to the software team—allowing software development to begin months earlier and preventing schedule slips in prototype delivery. The Virtual System Platform supports co-simulation with the Cadence Palladium® XP Verification Computing Platform and virtual prototype verification with the Cadence Incisive® Verification Platform.

### Early Software Development

Realizing today's mobile multimedia devices—with their increasing software content—on-time and on-budget requires starting software development with or before hardware components have been completed. Virtual prototypes have been employed to begin software development earlier than with FPGA or silicon prototypes. However, previous generation virtual prototypes do not support fully coherent multi-core hardware/software debugging, may not support open SystemC® models, and are difficult to create and support—all of which have limited their adoption.

The Cadence Virtual System Platform simplifies the creation and support of virtual prototypes with automated modeling and faster debugging. Design teams can begin developing software weeks to months before a hardware prototype is available, and software teams can use it as their application development platform.

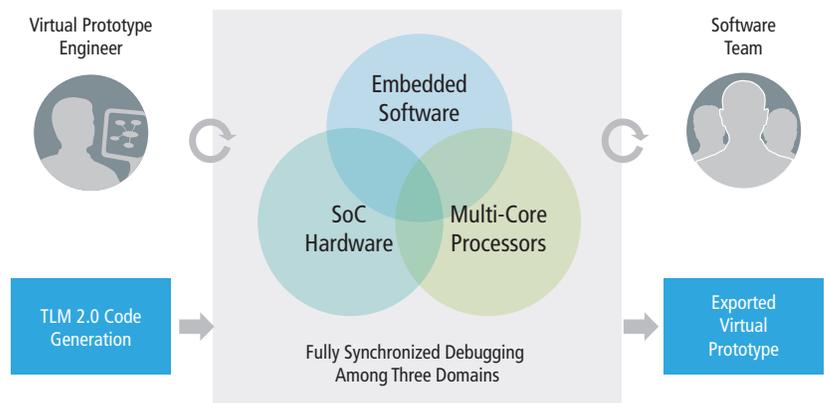


Figure 1: The Cadence Virtual System Platform includes all required hardware and software components to simplify virtual prototype creation with automated modeling and faster debug

### Easy Virtual Prototype Creation

Developing high-performance virtual prototypes has traditionally been difficult and time consuming. The Virtual System Platform automates the creation of virtual prototypes. Unlike handwritten models that take a lot of effort to type correctly, automated code generation reads an IP-XACT or text input and produces a transaction-level model (TLM) 2.0 template with

embedded register-intent awareness, and error checking for registers, without requiring TLM 2.0 knowledge.

A library of TLM IP models are available to incorporate into virtual prototypes. As SystemC source code they can be modified as needed for adapting to new requirements and tuning for performance or detailed functionality.

Fast processor models provide the high-performance execution speed required for an effective software development solution. The Virtual

System Platform supports ARM® Fast Models and Imperas fast processor models to provide hundreds of millions of software instructions per second (MIPS) execution and lock-step software and hardware debugging.

With TLM-aware analysis of bus bandwidth and resource utilization, the Virtual System Platform enables design teams to assess and optimize system performance and throughput.

## Fast Software Debug with Virtual Prototypes

The Virtual System Platform's unified debug GUI provides fully synchronized, coherent multi-core hardware/software (HW/SW) debugging. It comes with consistent breakpoints, single stepping, probing, tracing, and memory/register source-level debugging in either HW or SW models. Hardware debugging is based on a virtual platform-aware abstraction, built on a core of TLM-aware and SystemC debugging features.

The GUI itself is segmented and can be configured for the views most familiar to software or hardware engineers, or a combination of the two for efficient HW/SW debugging.

## Benefits

- Begin software development months before RTL and FPGA prototypes are available
- Create a first working virtual prototype in days versus weeks
- Improve collaboration with software development teams
- Rapidly debug complex HW/SW models
- Quickly bring-up silicon with working software

## Additional Features

### Deploy Virtual Prototypes for Software Development

The Virtual System Platform packages and exports the virtual prototype as a black-box-executable model for easy delivery to the software development team. The virtual prototype is commonly available

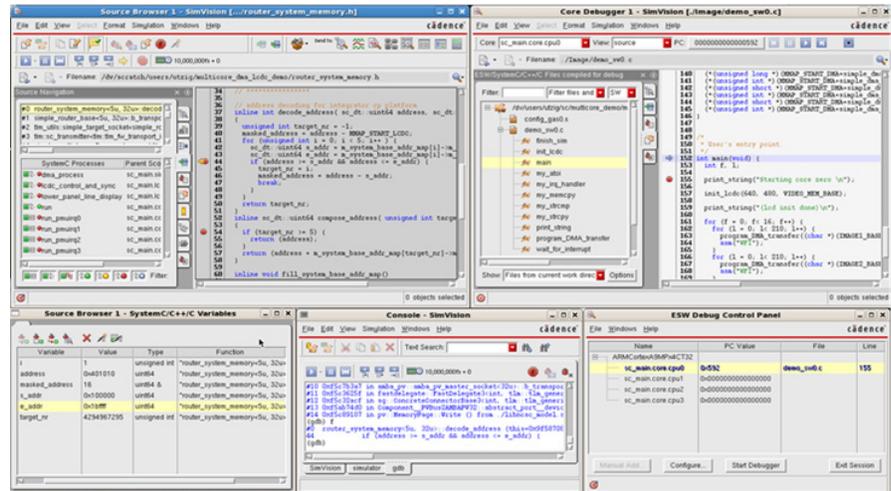


Figure 2: Coherent hardware/software debugging

months ahead of RTL- and FPGA-based prototypes. Preferred third-party compilers and debuggers (such as those from ARM, Lauterbach, or GDB) can be used with the virtual prototype to leverage it within existing software development environments.

The virtual prototype provides an economical and easily supportable complement to the FPGA-based prototype for software developers. An FPGA-based prototype enables verification of cycle-accurate HW/SW behavior. The virtual prototype is often higher performance and provides more controllable HW/SW debugging and replay.

Frequently, bugs discovered by software engineers cannot be debugged without visibility into the hardware. It can be difficult to reproduce and package the software environment and test case for debugging by the hardware team. The Virtual System Platform enables full-visibility HW/SW debugging at the software engineer's desk, allowing bugs to be addressed much faster.

### Connect to the Implementation Flow

Mixing RTL into virtual prototypes may be necessary for legacy IP, or as a means to verify cycle-accurate behavior for portions of the system. With the Virtual System Platform, a mixed-abstraction virtual prototype can be executed in the unified SystemC/RTL simulation engine

and debugger, or co-simulated with the Cadence Verification Computing Platform (Palladium XP).

Functional verification automation can be applied to the virtual prototype with embedded software, improving overall system quality by exploring corner-case system conditions that are often only discovered after RTL is used to build FPGA or silicon prototypes.

Creating RTL can be achieved manually, or by refining the TLM and using high-level synthesis with Cadence C-to-Silicon Compiler. The Universal Verification Methodology (UVM) can be applied to the virtual prototype and refined with each design refinement to find bugs earlier and streamline the overall verification effort.

## Components

### Design Input

- SystemC TLM 1.0 or TLM 2.0
- Legacy RTL: Verilog®, VHDL, or SystemVerilog
- Software: C/C++/Assembly

### Library of Models Available from Cadence

- Basic components: UART, keyboard/mouse controller, real time clock, programmable timer, interrupt controller, multimedia card, audio codec interface, programmable LED, color LCD, etc.

- Complex components: Ethernet controller, I2C, SPI, bus controller, serial interface, buffer, memory logger, battery, touch screen input, flash memory, initiator, multi-plexor, arbiter, router, etc.
- Virtualized hardware: terminal, Ethernet, etc.

### Supported High-Performance Processor Models

- ARM Fast Models
- Imperas OVP fast processor models for ARM, MIPS, Renesas, and PowerPC processor families

### Supported Software Debug Tools

- ARM DS 5, Imperas, Lauterbach, and other third-party debuggers

### Software Verification and Analysis

- Imperas CPU and OS-aware tools for software verification and analysis

## Requirements

### System Creation

- Linux workstation
- 64-bit Red Hat Enterprise or SUSE Enterprise
- 32GB of RAM

### Software Development

- Linux workstation (32-bit or 64-bit; Red Hat or SUSE)
- 2GB of RAM

## Cadence Services and Support

- Cadence application engineers can answer your technical questions by telephone, email, or Internet—they can also provide technical assistance and custom training
- Cadence certified instructors teach more than 70 courses and bring their real-world experience into the classroom
- More than 25 Internet Learning Series (iLS) online courses allow you the flexibility of training at your own computer via the Internet
- Cadence Online Support gives you 24x7 online access to a knowledgebase of the latest solutions, technical documentation, software downloads, and more



Cadence is transforming the global electronics industry through a vision called EDA360. With an application-driven approach to design, our software, hardware, IP, and services help customers realize silicon, SoCs, and complete systems efficiently and profitably. [www.cadence.com](http://www.cadence.com)