

INTEGRATING INSTRUMENTATION TOOLS INTO SYSTEM-LEVEL VERIFICATION FLOWS

DR. NEAL STOLLON, FS2 DIVISION, MIPS TECHNOLOGIES, INC.

```
vpi='h05  
vci='h1020  
hec='h39  
src port='h02  
seq number='h04  
description="cellReceived"  
error count='d0
```



In considering the overall verification flows for complex SoCs, the traditional EDA focus on pre-silicon analysis only addresses part of the problem. Simulation and formal verification approaches play an important role in the verification and initial integration of hardware designs and IP, but are more limited in terms of verifying the hardware/software system that characterizes most embedded processors. Unless high levels of abstraction are used (in which case important details may be lost) or significant amounts of computing resource and time are available, it is often not feasible to analyze processor subsystems operations over complex applications involving millions of operations and clock cycles using purely software solutions.

This need to address the verification of large complex systems over extended amounts of processing time is one widely used application of emulation and other hardware prototyping systems. But the price of hardware-based verification is a loss in the visibility of operations of internal signals. Especially in the case of processor systems, the ability to interactively control and understand operations requires dedicated instrumentation, not only at the processor(s) level, but also increasingly at the bus and co-processor or peripheral level.

Figure 1 shows this split of software-oriented hardware analysis and hardware-oriented software analysis that characterizes modern system analysis. "On-chip" instrumentation (OCI) provides significantly improved and simplified visibility into the internals of hardware operation and related software execution of embedded systems.

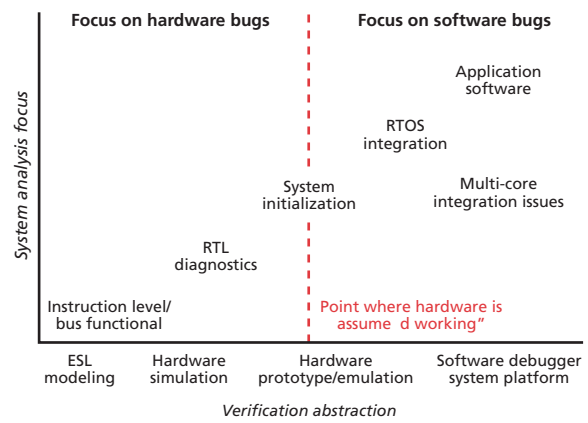


Figure 1: Different analysis tools for different tasks

FS2 is a leader in this emerging area of OCI: the integration of debug and trace IP for embedded processors and subsystems, along with supporting probe interfaces, and software tools for enabling on-chip control and visibility of embedded systems. Through our work with Cadence, FS2 is extending instrumentation-based debug into the emulation world by supporting the special requirements of debug interfaces for emulation systems interfaces.

As an example, emulation platforms may have slow or intermittent clocks that complicate the probe interface, especially for JTAG signals, which are commonly used debugger interfaces. FS2 and Cadence have implemented an adaptive clocking scheme for JTAG called RTCK that allows synchronization and correct JTAG sampling at higher speeds,

even with irregular emulator clock interfaces. The combination of support features for RTCK and for probing at arbitrarily slow clock speeds allow FS2 probes to support both Cadence® Xtreme® and Palladium® emulation systems and are being used by common customers for processor and system analysis today. (See Figure 2.)

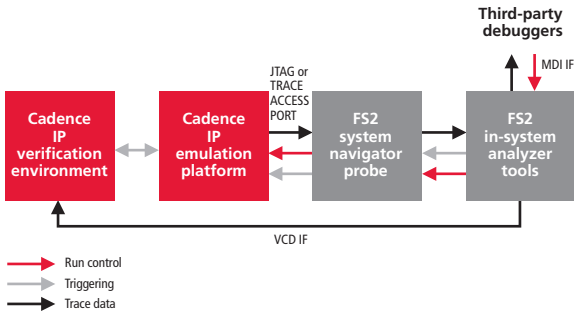


Figure 2: Cadence to FS2 interfaces

In addition to providing added value to the emulation platform, instrumentation-based debug provides useful information to simulation-based verification by allowing captured trace and hardware data to be compared to simulated data used in common VCD formats. This offers better insight and provides more reliable analysis on both the simulation and emulation sides of the verification environment.

The systems analysis loop that ties together EDA-based simulation and debug-based instrumentation into an end-to-end system verification flow allows for more consistent and comprehensive analysis than either provides alone. Since the same instrumentation IP can be simulated and then implemented in both the emulation environment and on the ASIC itself, OCI provides the valuable glue that allows a constant debug and analysis environment to be used throughout all stages of the design process. Related work can allow assertions developed at simulation to be extended and reused as triggers for instrumentation-based trace and related operations. (See Figure 3.)

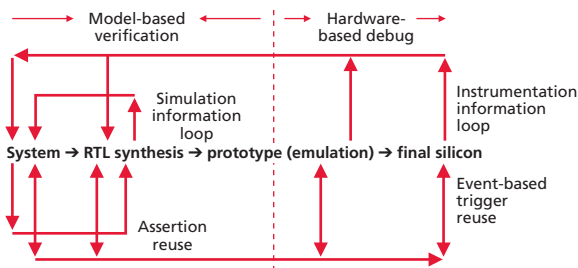


Figure 3: Verification and debug analysis loop

In looking at the differing types of on-chip instrumentation, they roughly break into four major types of functions.

1. **Core debug:** Most processor IP includes some debug blocks that simplify run control (go, halt, single step) and optionally provide instruction and data trace (MIPS™ EJTAG, PDtrace™). These core debug blocks and debugger features can differ significantly from processor to processor.
2. **Logic debug:** For more generic control and trace, IP that essentially allows the embedding of a logic analyzer on the chip provides major flexibility and visibility into the IP operation by enabling trace of deeply embedded signals.
3. **Bus debug:** Embedded bus fabrics such as AMBA and OCP present additional challenges for system debug due to complex interactions and the sheer amounts of data transferred over bus channels. These require instrumentation that addresses in-line bus filtering and complex triggering.
4. **System cross-triggering:** For multi-core systems, a comprehensive multi-core view and the ability to monitor events from different cores and to send control actions to different cores is required to synchronize and manage the complexity. Cross-triggering instrumentation provides an efficient and flexible means of controlling and coordinating the concurrent operations of several cores and IP.

See Figure 4.

These are basic instrumentation categories. The number of specialized and customized instrumentation components to address information such as system or core performance analysis is even larger. As important as the OCI function is itself, it is its integration and communication with other tools and user interfaces that makes it attractive. Many instrumentation systems utilize JTAG as a primary debug interface. Others use more specialized and higher performance debug access ports. *Both approaches are being used by Cadence and FS2 customers as part of their verification methodologies.* The ability to interface different instrumentation blocks seamlessly to different debug tools requires a sophisticated probe and instrumentation software environment that supports the requirement to serve differing and concurrent debug requests. FS2 provides a “debug cockpit” environment, including a user transparent multi-core API layer designed specifically to support the control of many interacting on-chip instruments and a corresponding higher level API (MDI) to control the software-level interfaces.

The need to verify more complex SoCs requires new ways of thinking about all the analysis tools that are available; to gather and integrate information from all the views and abstractions that are available. EDA verification and embedded hardware debug have traditionally been considered different domains, but the analysis requirements for processor-centric multi-core systems are driving them much closer together. On-chip instrumentation (OCI) IP and tools are one approach to bridging these analysis approaches into the overall customer system verification environment and providing a debug cockpit that can support different stages of design analysis in a consistent manner. (See Figure 5.)

Increasingly, OCI is used to address the real-time analysis needs of the interactive hardware and software of embedded design. With proper implementation and integration, instrumentation of a design can provide a set of tools that can span the verification flow of a design, from system simulation to emulation prototyping and into final silicon. By providing a consistent set of information across these stages and their diverse analysis requirements, instrumentation adds to the flexibility and cohesiveness of design views for both software verification and hardware debug.

FOR MORE INFORMATION

Email us at info@cadence.com or visit www.cadence.com.

Neal Stollon is a systems engineer and Director of Technical Marketing with the First Silicon Solutions Division of MIPS Technologies. He has more than 25 years of digital design and processor development experience at Texas Instruments, LSI Logic, Alcatel, and others. Dr. Stollon has a Ph.D. in Electrical Engineering from SMU, is a Texas Professional Engineer, has written more than 30 technical papers, and holds 7 patents.

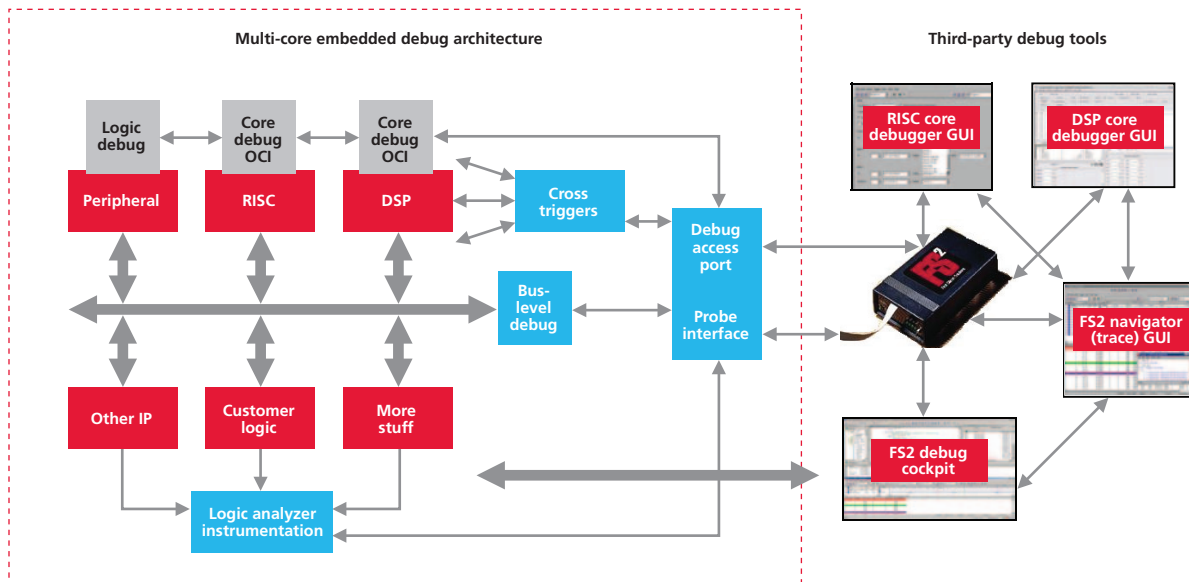


Figure 4: A fully instrumented multi-core design

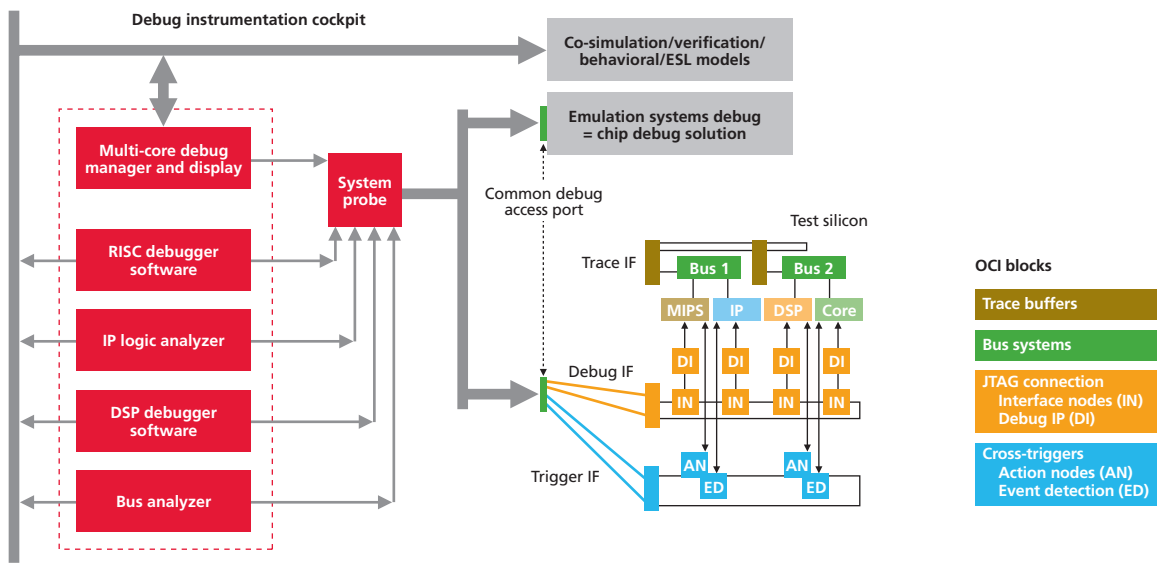


Figure 5: A debug cockpit approach spanning different instruments, debug resources, and modeling views