

# The Hard Facts on Hardware-Assisted Verification

Learn more about present and future of hardware-assisted verification—namely emulation and prototyping—which is becoming critical to creating the next generation of IC designs.

**H**ardware-assisted verification (HAV) systems are a must-have for designing next-gen systems-on-chips (SoCs) and the systems that surround them. Even though emulation and prototyping have been part of the fabric of the electronics industry for decades, lots of myths remain about what they can do, how to implement them for chip verification versus software validation, and how they're evolving.

It's time to dispel the biggest misconceptions about the technology:

## Hardware emulation and FPGA prototyping offer the same features, functionality, and benefits.

No. Hardware emulation and FPGA-based prototyping have distinct attributes and are implemented differently for SoC and software verification and validation.

Emulation, with its ease of compiling hardware-description-language (HDL) code into an executable model and its full observability of signals, is used for register-transfer level (RTL) design and verification. FPGA-based prototyping achieves greater execution speed than emulation in exchange for less flexibility and observability. It's a vital tool for software developers trying to validate their code against the RTL design.

**All hardware emulators, including FPGA-based prototyping tools or hardware-assisted verifications**

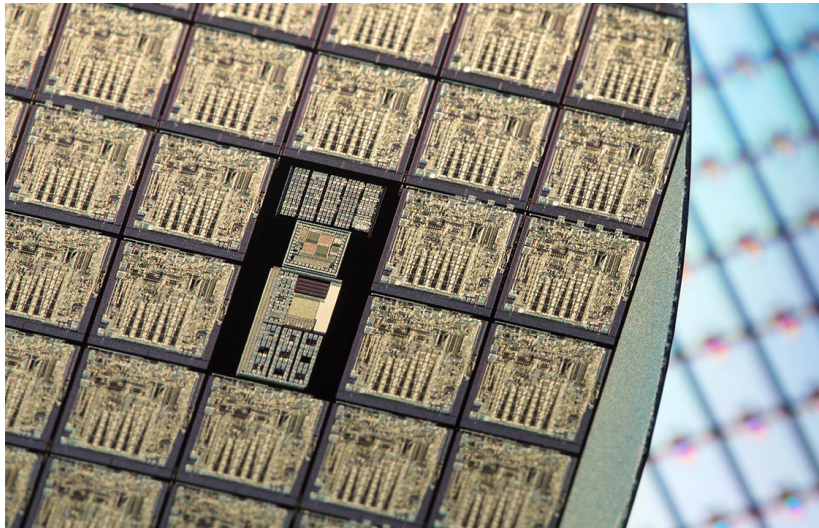
**platforms, are alike.**

[Hardware emulators](#) are based on purpose-built SoCs or large FPGAs that, depending on the vendor, may be custom-designed or sourced from a third party. These chips possess different attributes that endow emulation platforms with distinct capabilities, particularly verifying and debugging [the interaction of hardware with software in SoC designs](#) unlike any other solution.

Prototyping tools depend on different FPGAs with vary-



The three-part Siemens Veloce CS system spans (from top) emulation, hardware prototyping, and software prototyping. Image credit: Siemens



Hardware-assisted verification tools are evolving to handle the rising complexity of modern chips, including 3D ICs. Image credit: Siemens

ing performance metrics based on the vendor. Prototyping has long been and continues to be widely used in software development to accelerate the development of firmware, operating systems (OS), and applications, while it can also lend a helping hand with system integration tasks. Modern innovations have made it possible to create a scalable, system-wide, [FPGA-based prototyping](#) system with a capacity ranging from 40 million gates to over 40 billion gates.

The latest hardware-verification systems continue to evolve. In some cases, systems can pack enough of a punch to integrate hardware emulation, enterprise prototyping, and at-speed software prototyping in a single system.

Such a level of integration offers congruency, speed, and modularity, accelerating verification and validation cycles and reducing the cost per verification cycle. For instance, at the heart of the latest [Veloce CS system](#) from Siemens is a purpose-built accelerator chip for emulation and AMD's FPGA-based adaptive SoC for enterprise and software prototyping.

### FPGA prototyping can replace hardware emulators.

Unfortunately, no. Hardware emulators and FPGA prototyping are complementary with specific functions in a verification flow.

Hardware engineers and software developers tend to choose HAV systems for a specific task since each task has unique requirements to enable faster time to project completion and decrease cost per verification cycle. For example, logic designers use an emulation platform to integrate IP and take care of chip-level verification. Software developers attempt to boot the OS on the chip design using the prototyping platform. Both are often used for different facets of

the same project.

### Hardware-assisted verification platforms have not actually changed much in recent years.

The chip design landscape is changing rapidly due to the vast scale of the latest designs and the demands of data-center CPUs and GPUs as well as [purpose-built AI accelerators](#). The rising complexity of software stacks is also throwing chip engineers for a loop. In many use cases, today's chip isn't the product. Instead, the product is a complete system with a full software stack that designers must verify.

HAV platforms are changing and adapting to meet these new verification and validation requirements. In some cases, these systems are unifying hardware emulation, enterprise prototyping,

and software prototyping to accelerate verification and validation cycles. This gives RTL designers, software coders, and system-level engineers a way to effectively collaborate and communicate using one common user interface, RTL models, and database.

Each system meets the demands of its role in the design process with distinct attributes and is implemented in a different way.

### Hardware/software co-verification is impossible to execute on a modern HAV platform.

Today's [chip designs are huge](#), and the interactions between the chip and the system can be a challenge to unpack.

Today, verification involves more than verifying the RTL of the chip design. You need to validate the rest of the system surrounding it, and that requires the verification of interactions between the chip, the circuit board, and the mechanical subsystems within the overall design. The other piece of the puzzle is proving correct operation of the full software stack often with application code.

Projects of this scale require [hardware, software, and system-level co-design and co-verification](#). That's why a project team needs an emulation platform, enterprise prototyping platform, and an at-speed prototyping platform.

### Hardware-assisted verification is having a hard time meeting the demands of AI, data center, high-performance computing (HPC), and embedded software applications.

Hardware emulators and prototyping systems have had to evolve to meet the demands of the latest leading-edge designs. Several commercially available HAV systems effectively address all of them. HAV platforms are now [a must-have for verification and validation](#) of these complex sys-

tems, executing a diverse range of workloads.

### **Systems companies are rewriting the chip design and verification playbook—one that leaves out HAV.**

Untrue. Systems companies are designing purpose-built SoCs, and they have different expectations than traditional IC design companies. The product isn't the chip itself. It's the in-house chip placed on its in-house circuit board design running its entire in-house software stack from drivers and operating systems up through applications. The whole integrated system must be tested—and not just modules of the RTL code. These SoCs are designed to execute specific workloads software effectively and deliver performance and power gains that they're apparently not seeing from their chip vendors.

As these companies look to stay relevant in the AI age, they're pushing into the latest process technologies and adopting very large chiplet-based systems. In that context, hardware-assisted verification is becoming key.

### **HAV tools lack the congruency, shareability, scalability, and speed features required to handle today's large and complex designs.**

Not true. Many of today's HAV systems are engineered for all of these features, giving engineers access to the right tool for the right task.

Congruency is more than a vague concept. It's implemented across integrated HAV systems so that an RTL model will produce the same behavior wherever it runs, just at different execution speeds and levels of observability. Congruency is another piece of the puzzle, since it enables you to move seamlessly between HAV platforms.

Shareability, also a standard feature in HAV systems, is necessary for the overall enterprise rather than an engineering group using the systems. As enterprise-class assets, these systems should be able to support multiple groups or design projects worldwide. They reside in the secure networks of a data center.

Scalability is a critical feature for many modern verification systems, too. Since leading-edge designs, whether monolithic or multi-die, can comprise tens of billions of gates, it's critical to deliver maximum capacity and granularity.

Performance is non-negotiable, because execution speed and compile speed depend on it. The latest verification tools are being upgraded to run long and very complex workloads not only for functionality, but power-performance analysis.

### **Hardware-assisted verification can stumble when attempting to verify chips with enormous gate counts at advanced process nodes.**

This could turn out to be true depending on the age of the system. But the next generation of hardware-assisted verification tools can handle chip designs packing 40 million up to more than 40 billion logic gates at a time. These platforms

are able to execute full system workloads with visibility and congruency using the right tool for the task for faster time to project completion, while decreasing the cost per verification cycle.

### **The emerging market for chiplet verification will pass by HAV platforms.**

Chiplets are an incredible opportunity to design a complex system mixing and matching a wide range of different functions, not just logic.

It's also an opportunity for providers of HAV platforms. In a way, verifying a chiplet-based design isn't all that different from verifying several distinct chips spread out around a system. It can be challenging, though, because all of the chiplets are placed unusually close together in the package, using high-density die-to-die interconnects to chat with each other over SoC fabrics.

The other issue with chiplet verification is related to capacity. These multi-die systems aren't bound by the reticle limit of a single monolithic SoC. In many cases, they cram more than a thousand square millimeters of silicon in a single package, which can be hard for an HAV platform to wrap its head around. It's a capacity scaling and reliability challenge.

Not all hardware emulation and prototyping systems have enough capacity to handle these huge chiplet-based designs. But several do.

### **Power analysis continues to thwart HAV platforms.**

Not at all. In fact, you could argue that hardware-assisted verification is essential for [power analysis](#). Power analysis, optimization, and management has become as important—maybe even more so—than performance and area, the other key variables used to measure advances in chip designs.

With hardware-assisted verification, it's possible to get visibility on all the signals in a design and extract this information for power analysis. Since today's systems have the capacity to handle tens of billions of logic gates at a time, it's possible to consider a SoC design in its entirety.

HAV enables power engineers to “shift-left” and perform power analysis at the SoC level even earlier in the design process. They can test the SoC with real software workloads and take corrective action in software or hardware.

*Vijay Chobisa is a senior director of product marketing for hardware-assisted verification at Siemens with more than 20 years of experience in the world of hardware emulation. Chobisa holds a Bachelor's of Science in electronics and communications engineering from Jai Narayan Vyas University, Rajasthan, India.*

