

Software-Defined Hardware-Assisted Verification for the AI Era

Ushering in scalable emulation and prototyping on stable hardware platforms with continuous SW innovation across all use cases

Author

Frank Schirrmeister

Executive Director
Strategic Programs
System Solutions, Synopsys

Executive Summary

Semiconductor verification is under unprecedented pressure. Designs are bigger and more modular, are distributed across chiplets in multi-die designs, interface standards evolve faster, and above all products are defined by the software they run.

As development teams need to validate complex software stacks and AI workloads pre-silicon, Hardware-assisted Verification (HAV)—emulation and FPGA based prototyping—has become the new keystone for validating functionality, power, and performance running quadrillions of verification cycles across IP, subsystems, SoCs, and multi-die systems.

As changes to the actual hardware of these HAV platforms only happen every couple of years, software has now become the force multiplier for those platforms, delivering continuous improvements in throughput, debug, and use case support over the lifetime of the same hardware platforms.

This white paper introduces Synopsys' perspective on software-defined HAV. Synopsys defines it as the convergence of specialized hardware—the server oriented ZeBu Server 5 platform for the biggest data center AI training and computing designs, and EP-Ready hardware modules for mainstream designs in ZeBu-200 and HAPS-200 platforms—with flexible software stacks supporting synchronous clocking for emulation use cases and asynchronous clocking to offer the ultimate best performance.

Synopsys articulates three layers that make software-defined HAV real: (1) Hardware Platform Specialization, with Synopsys EP-Ready Hardware Architecture, (2) Software Stack Flexibility and Upgradability, and (3) Use Case Enablement. The result is a lifecycle in which software updates add capability—performance, debug, max capacity, and new use cases—without changing the underlying hardware.

Synopsys announced, at Synopsys Converge Conference 2026, the next expansion of its HAV portfolio:

1. **Software-defined HAV:** Software increases performance on existing hardware and expands use cases, improving lifetime value. EP-Ready Hardware enables all emulation and prototyping use cases. Modular HAV scales to mega designs exceeding 50 Billion Gates for ZeBu Server 5 and hybrid setups.
2. **New Engines Join the Family:** HAPS-200 12 FPGA and ZeBu-200 12 FPGA introduce a new EP-Ready base module with 12 of the latest AMD FPGAS, complementing the already existing EP-Ready 6 FPGA modules introduced in 2025.

3. **Capacity Improvements for the AI era:** The sweet spot for mainstream designs—the broad design base of mobile, client, server, consumer, edge AI designs as well as datacenter subsystems—moves from 6 FPGA to 12 FPGA designs. The 6 FPGA systems support sweet spot sub-systems, mobile/client/server cores, MCU, and consumer designs. Using Synopsys' unique Modular HAV technology, ZeBu Server 5 extends to much larger configurations for mega designs supporting data center AI (Training and Inference), GPU, custom accelerators, and networking (IPU/DPU).

Synopsys continues to lead with the biggest and the fastest HAV platforms—and with software that keeps making those platforms better over time. Verification success in the AI era is no longer just about raw hardware capacity, it requires software defined scale that continuously expands performance, debug, and use cases on the same HAV footprint.

The New Complexity in Semiconductor Design

Across datacenter, automotive, and edge systems like autonomous vehicles, industrial applications, and physical AI, more system companies are building custom silicon to meet stringent power, performance, latency, and software compatibility goals. Monolithic SoCs are giving way to multi-die designs and chiplet architectures, compounding the number of compute fabrics, memory hierarchies, interconnects, and interface protocols that must be validated together—often with full software stacks running pre silicon.

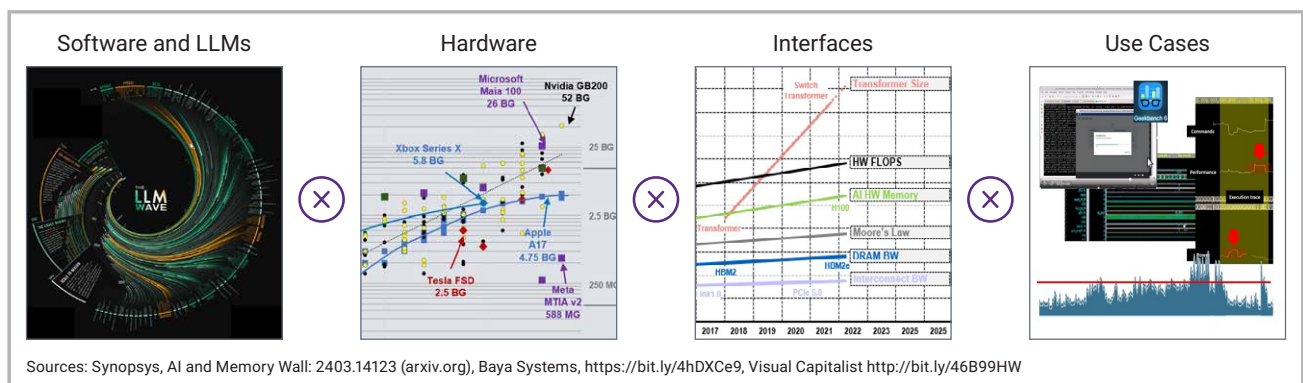


Figure 1: Four Compounding Complexities for Semiconductor Verification

Four compounding complexities now define verification requirements:

- **Software and LLMs:** Products are defined by workloads—training vs. inference, vision vs. text vs. multimodal—each imposing distinct KPIs (throughput, latency, energy per task). Software and AI workloads must be brought up pre-silicon, at speeds high enough to expose real integration and performance issues.
- **Hardware scale:** From subsystems to multi-die SoCs, top end designs push into the domain of tens of billions of gates and require long running scenarios to validate performance and power under realistic conditions. Subsystems of the largest datacenter designs are as big as full mainstream designs in mobile and consumer applications.
- **Interfaces:** PCIe®, HBM, LPDDR, Ethernet, and UCIe for die-to-die and many other protocols evolve faster than hardware refresh cycles, demanding continuous updates to enable at-speed, standards aware validation in the HAV environment.
- **Use cases:** Verification has broadened from RTL correctness into software bring-up and hardware/software validations, power/performance analysis, compliance/certification, safety/security, and hybrid virtualization—each with distinct requirements for MHz, visibility, I/O realism, and analytics. These must be supported pre-silicon to avoid late surprises.

Pure simulation remains indispensable for early functional checks and very detailed debug, but it cannot alone reach the quadrillions of cycles required to validate the full software experience. Emulation and prototyping deliver orders of magnitude speed-up, deep visibility, and real/virtual I/O to close that gap—supporting a "shift left" strategy in which software validation becomes a first class pre-silicon activity.

Market Forces and Verification Challenges

AI pressure is relentless. Workloads and models evolve quickly with software complexity growing even within specialized tasks. Verification must ensure that silicon meets workload specific KPIs (e.g., tokens/sec, images/sec, latency) pre-silicon, across realistic I/O and OS stacks, to avoid schedule slips and re-spins.

Meanwhile, interface velocity (e.g., PCIe generational ramps) and hardware scale (multi-die, chiplets) multiplies partitioning and coherency scenarios. The expanding set of use cases—from RTL regressions to multi-month AI workloads, from compliance to hybrid virtualization—drives the need for breadth and software defined evolution on the same platform.

Traditional flows hit hard limits of simulation throughput, late software bring-up, and siloed platforms that strand capacity.

The path forward has two imperatives:

- **Software-defined evolution:** continuous improvements in performance, debug, compile, bring-up, and use case coverage—on the same hardware
- **Scale and flexibility in HAV platforms:** capacity for very large designs, multi-unit scaling, robust reliability, and broad I/O options for realistic stimulus

The Synopsys HAV Portfolio—Hardware and Software Convergence

Synopsys has re-architected its HAV portfolio to support end to end demands—from early RTL debug to full software bring-up and performance validation for AI workloads, and new use cases like pre-silicon software-development for mixed-signal components and fault emulation for safety-oriented applications:

ZeBu Server 5: Capacity density and scaling for very large designs, modular HAV across many units, high reliability for long workloads, and best Total Cost of Ownership (TCO) in data center deployments.

ZeBu-200 (AMD Versal™ Premium VP1902): High emulation performance for large designs with faster compile and improved debug vs. prior generations, optimized for modular designs and multi user access. It now comes in two base-module configurations supporting 6 and 12 FPGAs, respectively.

HAPS-200 (AMD Versal™ Premium VP1902): High speed and at-speed prototyping for real world software bring-up, interface validation, compliance, and performance benchmarking. Supporting 6 FPGA and 12 FPGA base modules, HAPS-200 supports asynchronous clock domains and fast protocol adapters.

EP-Ready Hardware underpins ZeBu-200 and HAPS-200: A shared base module and cabling approach that allows teams to reconfigure between emulation and prototyping, enabling all emulation and prototyping use cases. As projects evolve, the same physical modules are reused across teams and phases, improving ROI and utilization.

Synopsys complements hardware leadership with software stacks that define bring-up speed, debug visibility, and workload throughput: ZeBu® Software for synchronous emulation and HAPS® ProtoCompiler for speed optimized partitioning and asynchronous prototypes. Both can be combined with Synopsys Virtualizer™ for hybrid flows that blend virtual models with HAV engines (e.g., for fast OS boots and mixed fidelity setups).

The Layers of Software-Defined HAV and Modular Methodology

Layer 1—Hardware Platform Specialization and EP-Ready Architecture

Synopsys HAV platforms (ZeBu Server 5, ZeBu-200, HAPS-200) are engineered for scalable capacity, modularity, and I/O richness, supporting very large designs or many concurrent jobs with high uptime. In emulation, predictable, synchronous execution supports multi-user debug and regression throughput. In prototyping, direct connect cabling and asynchronous clocking maximize MHz for extended software runs and interface validation.

EP-Ready hardware ties these modes together for ZeBu-200 and HAPS-200, allowing organizations to reconfigure hardware for either best design performance (prototyping) or best design flexibility (emulation) without changing the base system.

Technically, EP-Ready hardware centers on a base module (up to 12 AMD Versal Premium VP1902 adaptive SoCs) assembled with hubs/cables for either emulation or prototyping. Interface protocol solutions, memory/transaction models, and speed adapters connect the DUT to a realistic environment. The same physical systems can be repurposed across projects, from RTL verification to software bring-up to compliance/certification, aligning capital spend with evolving needs.

This specialization is not only about building bigger systems, but also about optimizing capacity and topology per use case and then reusing assets as needs shift. Multi-unit modular HAV provides the capacity and reliability required for months long workloads to avoid catastrophic recompile cycles. EP-Ready flexibility enables platform consolidation and future proofing.

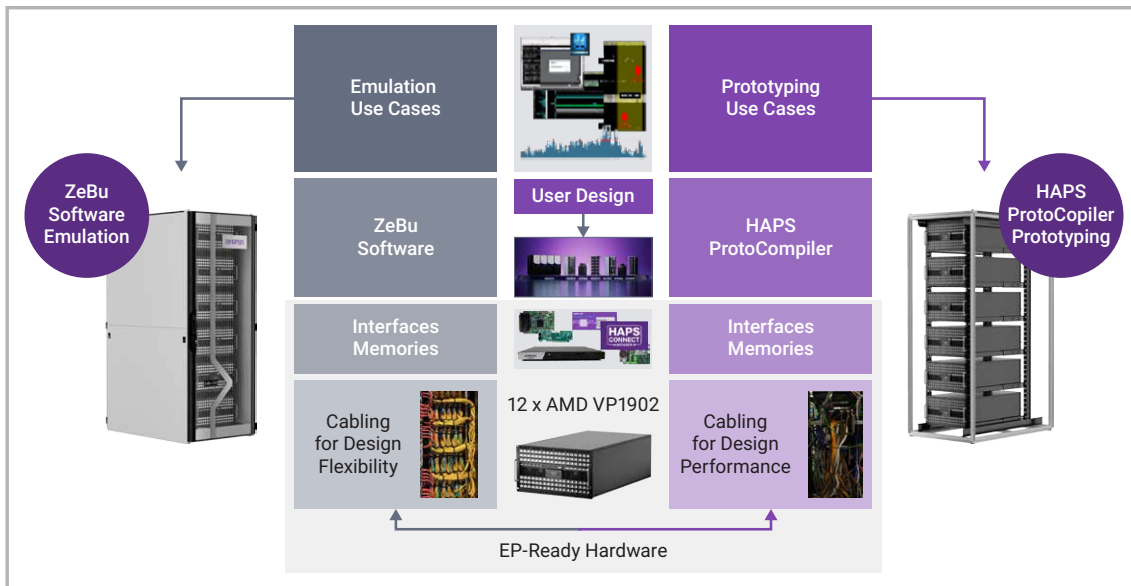


Figure 2: EP-Ready Hardware with Software-defined Use Cases

Layer 2—Software Stack Flexibility and Upgradability

In a software-defined paradigm, the software stack is the engine of continuous improvement. Synopsys invests in compile, runtime, debug throughput, and hybridization enabling more use cases on the same hardware. Over time, software updates have delivered up to 2× performance improvements and up to 3× faster time to model (TTM) since 2023 on existing platforms, with additional debug/readback/coverage advances—further raising throughput without hardware changes.

- **ZeBu Software** orchestrates synchronous emulation across large partitions with rich visibility; ongoing improvements span compile, trace memory handling, power analysis flows, and multi user scheduling.
- **HAPS ProtoCompiler** optimizes partitioning and clocking for asynchronous prototypes to reach higher MHz and more predictable long runs.
- **Virtualizer™** enables hybrid execution (fast OS boots, mixed fidelity) to accelerate milestones while hardware partitions mature.

The implication is strategic—**lifetime value** rises as software unlocks more capability—extending performance, debug, and analytics (power/perf/cache coherency) on installed hardware and simplifying roadmap transitions.

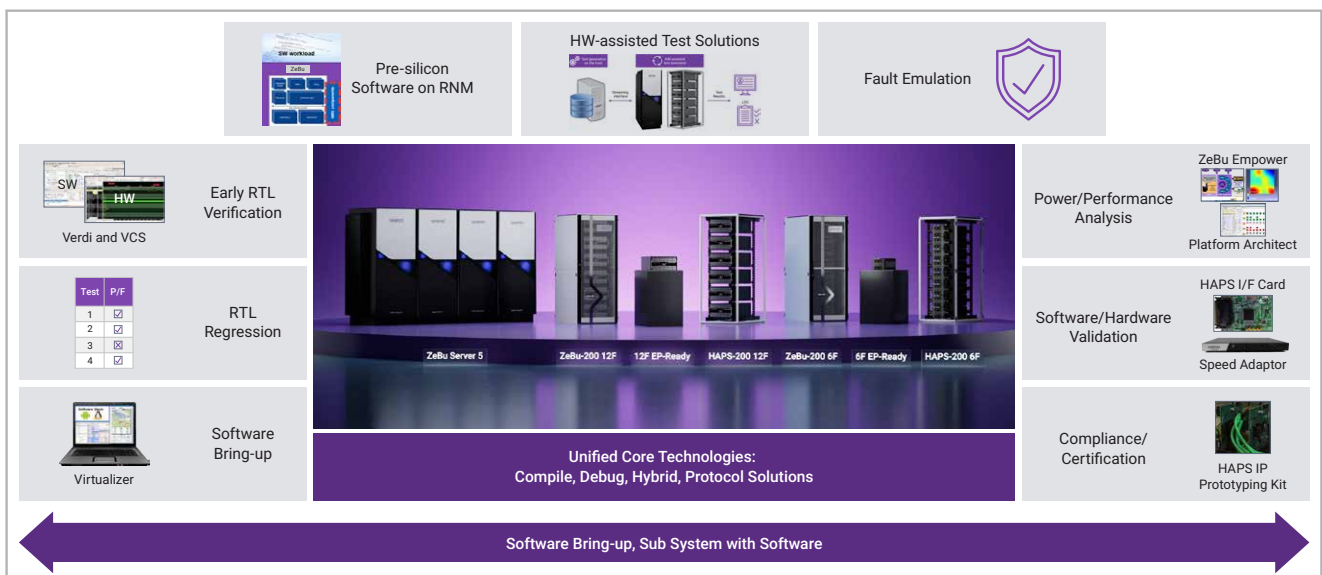


Figure 3: Use Cases across the Synopsys HAV Portfolio

Layer 3—Use Case Enablement

Layer 3 operationalizes software defined HAV.

It expands verification scenarios across phases and teams with use cases from early RTL verification and RTL regressions to software bring-up that often extends to hybrid execution blending virtual models with HAV engines.

The most recent releases extend the well adopted use cases for power and performance, software/hardware validation using interface speed adaptors and interface cards as well as compliance/certification with new pre-silicon use cases for hardware-assisted test automation, software development for mixed-signal components and fault-emulation to support safety and reliability analysis.

Key use cases beyond classic early RTL verification and RTL regressions include:

- **Hardware-assisted Test Solutions:** Building on Threadmill, Imperas DV and Sting technology, Hardware-assisted Test Solutions target verification of complex, many-core designs based on leading-edge CPU architectures with advanced stimulus generation and checking tools for the most difficult verification problems. High-level, portable test descriptions are executed on emulation at meaningful performance, allowing teams to stress corner cases for processor and coherency validation, and observe system behavior long before silicon. With Hardware-assisted Test Solutions, users reduce late surprises, improve verification quality,
- **RNM (Real Number Modeling):** For software development in the context of mixed-signal and system-level designs, RNM emulation enables fast, scalable abstraction of analog behavior within digital-centric verification flows. By modeling analog blocks with real-number precision, teams validate control algorithms, calibration logic, and cross-domain interactions at orders-of-magnitude higher speed than transistor-level simulation. The outcome is earlier architectural insight, faster software bring-up, and confident system validation without sacrificing correlation to silicon intent.
- **Fault Emulation:** For safety-critical and high-reliability designs, the fault emulation use case enables scalable fault injection and analysis across RTL, emulation, and prototyping. Accelerated fault campaigns allow teams to evaluate diagnostic coverage, safety mechanisms, and error recovery under realistic workloads, rather than synthetic tests. The result is improved confidence in functional safety compliance, reduced verification cycles for fault scenarios, and predictable progress toward certification milestones.
- **Software/Hardware Validation:** Targeting MHz-level performance on mid sized designs, gains realized through speed adapters, optimized partitioning, and improved orchestration on emulation/prototyping accelerate bring-up and perf/power analysis milestones relative to prior baselines.
- **Emulation Modularity for Hyperscale Compute:** For very large gate count and long AI workloads, Modular HAV in emulation allows to mimic the modularity of large scale SoCs and multi-die design, scaling to dozens of units connected using the actual design interfaces (like AXI) across connections between the ZeBu and HAPS platforms. The outcome is reliable performance validation at scale and better schedule predictability.
- **Multi-die Modular Prototyping Methodology:** With Modular HAV prototyping approaches, users prototype individual dies/subsystems first then integrate at the system level using standard interfaces and asynchronous clocking. They can combine this with virtual prototypes when helpful to further balance emulation and prototyping hardware utilization with virtualized representations of processor sub-systems.
- **Compliance and Certification:** Standards compliance and real workload validation benefit from prototypes running at tens of MHz. Asynchronous clocking and interface adapters sustain long tests, reducing the wall clock time for compliance and performance sign off.

Across these use cases, three themes consistently emerge: Teams achieve faster bring-up and greater predictability by combining MHz-level FPGA prototypes with the rich visibility of emulation-class debug. They maintain reliability over weeks-long, software-realistic workloads through robust multi-unit orchestration. They maximize ROI with EP-Ready reuse, avoiding stranded capacity and dynamically reallocating the same hardware between emulation and prototyping as project needs evolve.

The 2026 Synopsys Hardware-assisted Verification Portfolio

Since 2023, software upgrades have delivered up to 2x performance improvements and up to 3x faster Time to Model, while readback and coverage enhancements, together with broader debug throughput, raise day to day verification productivity. The 2026 portfolio emphasizes software defined gains on existing hardware alongside new 12 FPGA systems and expanded modular capacity.

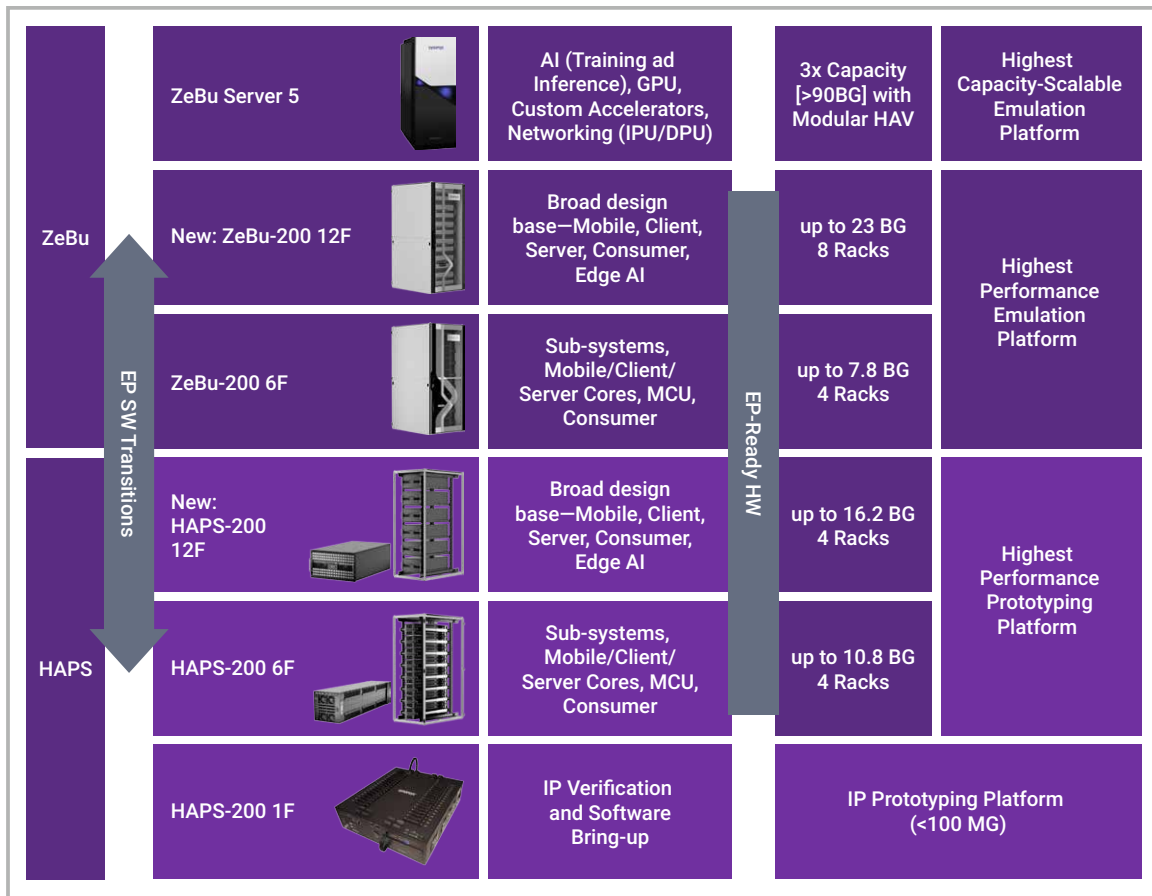


Figure 4: The 2026 Synopsys Software-defined HAV Portfolio

Along with continuous software improvements, two new engines, HAPS-200 12 FPGA and ZeBu-200 12 FPGA, extend the mainstream sweet spot capacity beyond the prior 6 FPGA class, adding capacity headroom and throughput to support faster bring-up, compliance, and sustained software workloads. At the high end, modular HAV scaling on ZeBu Server 5 pushes mega design capacity roughly 2x, enabling months long AI workloads to run reliably with robust multi unit orchestration.

The new hardware platforms in the HAV Portfolio scale to cover IP, subsystems, mainstream SoCs, and the lower tier of mega programs, while preserving EP-Ready reconfiguration between emulation and prototyping to maximize utilization and ROI.

- For the largest, datacenter class designs, users prioritize **ZeBu Server 5** configured with **modular HAV**, where capacity headroom, long run reliability, and robust multi unit orchestration are essential to sustain weeks long, software realistic workloads as well as validate performance at scale.
- For **mainstream SoCs** across mobile, client, server, consumer, and edge AI, and for data-center subsystems, users apply the **ZeBu-200 12F** and **HAPS-200 12F** systems to expand the sweet spot beyond 6F. Begin with emulation to accelerate early RTL coverage and power analysis, then move into prototyping for MHz level software bring-up, interface compliance, and performance validation. Users can switch between modes via **EP-Ready** reconfiguration as the project matures.
- For **subsystem/IP verification and software enablement** at smaller capacities (from <~100 MG upward), users deploy **HAPS-200** in compact footprints (including 1F variants) to speed IP bring-up, driver development, and interface validation, leveraging protocol adapters and IP prototyping kits to shorten integration cycles. At the same time, they preserve the option to pivot into emulation when deeper visibility or synchronized debug is required.

How Software-Defined HAV Changes the ROI Equation

Software defined HAV is about more than raw speed; it's about **compounding gains** on the same hardware footprint:

- **Performance and TTM** gains (up to 2X and 3X, respectively) on installed systems translate into more compile turns/day and faster iteration cycles, crucial for both RTL regressions and software bring-up.
- **EP-Ready** reuse stops the “single purpose rack” problem. The same hardware toggles between emulation and prototyping workflows as needs evolve, driving higher asset utilization and increasing ROI.
- **Modular HAV** on ZeBu Server 5 unlocks very large designs and long workloads, with its reliability safeguarding weeks long runs from failure, an operational necessity for AI workloads.
- **New use cases** (e.g., HAV Test Solutions for cache coherency and subsystem verification, RNM and fault emulation) reflect the Synopsys approach of adding software enabled capability between hardware refresh cycles, keeping the platform relevant as protocols and verification scope evolve.

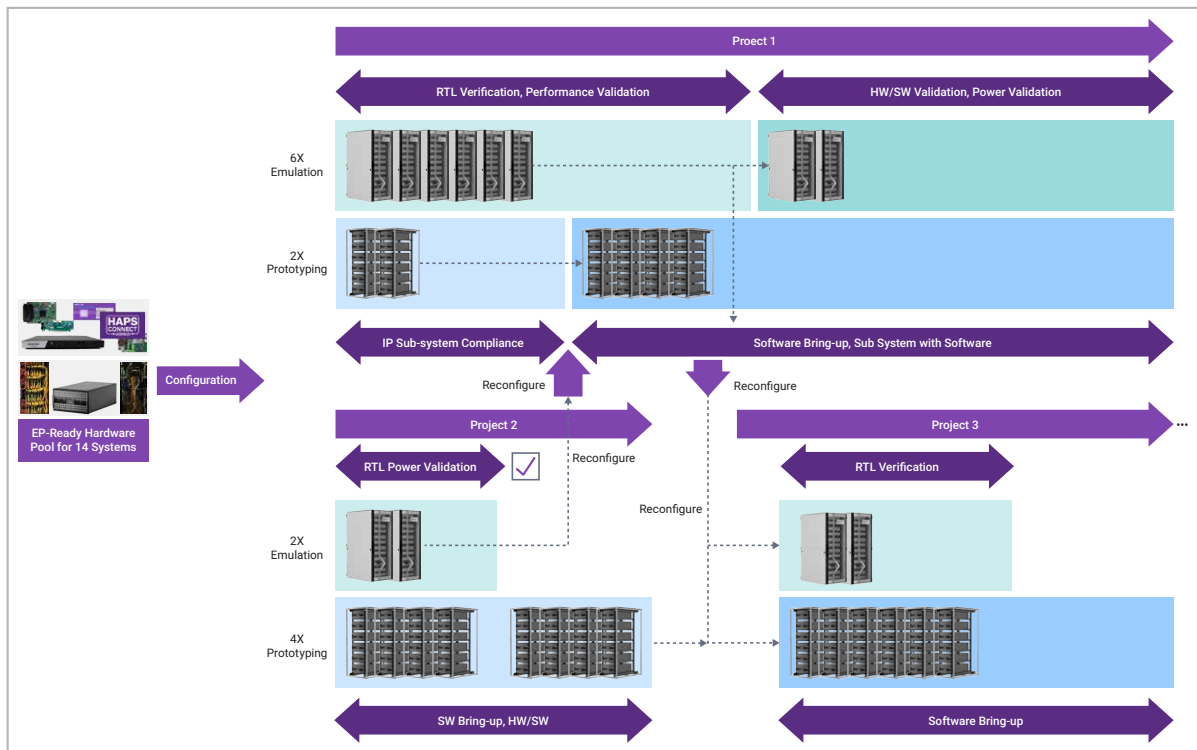


Figure 5: The EP-Ready Hardware Across a Project Flow and Use Cases

Conclusion and Outlook

The AI era compresses schedules, amplifies complexity, and raises the stakes for first-pass silicon success. Software-defined HAV responds with a strategy that compounds value over time. Specialized hardware—ZeBu, HAPS, and ZeBu Server 5—delivers the scale, I/O richness, and reliability demanded by modern verification, while the ZeBu Software, HAPS ProtoCompiler, and Virtualizer stacks drive continuous improvements in performance, time to model, debug, and hybridization on the very same systems.

The EP-Ready architecture turns racks into multi purpose infrastructure, enabling organizations to fluidly transition between emulation and prototyping without duplicative capital. It also allows teams to align assets with a fast moving spectrum of use cases—from early RTL regressions to long running AI workloads, power/performance analysis, compliance and certification, and hybrid virtualization.

For engineering leaders, the practical mandate is to consolidate and future proof on EP-Ready platforms and lean into software updates to expand throughput and analysis depth across the installed base. At the same time, they must adopt modular and hybrid methodologies that manage multi-die complexity while bringing realistic software workloads into the pre silicon phase.

With the largest, fastest HAV platforms and targeted capacities spanning datacenter class to mainstream designs and down to subsystems and IP—continuously improved through software—Synopsys enables teams to shift left, scale validation with confidence, and deliver robust silicon on time.

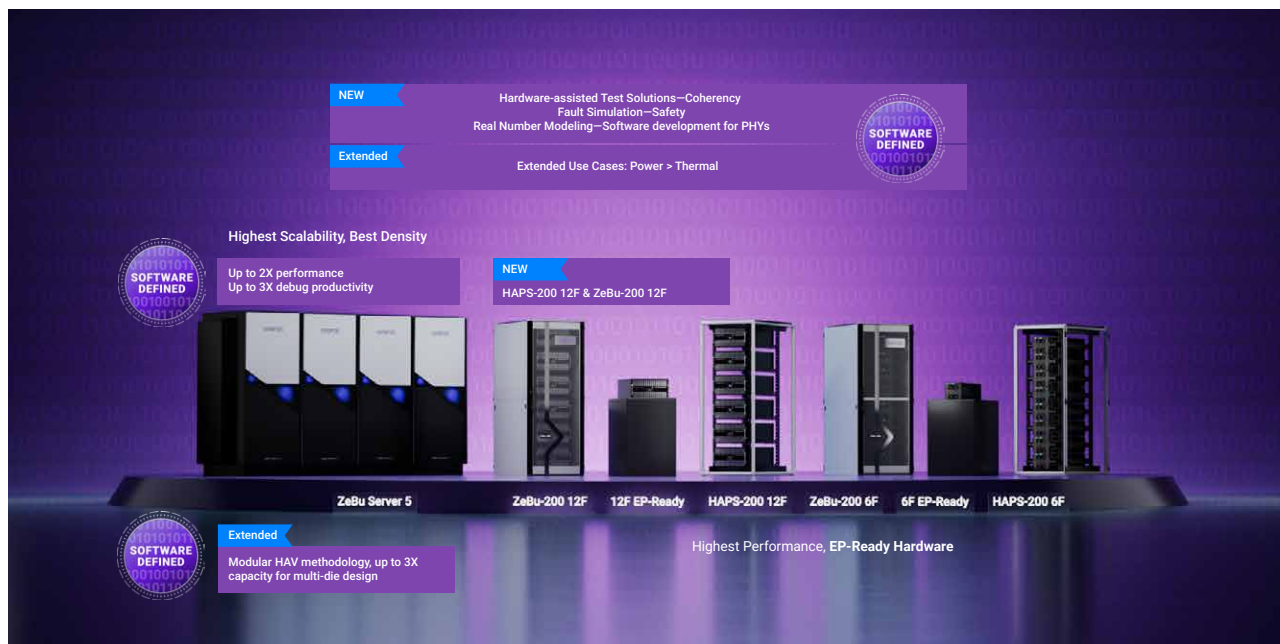


Figure 6: The 2026 Synopsys Software-defined HAV Portfolio