

## DATASHEET

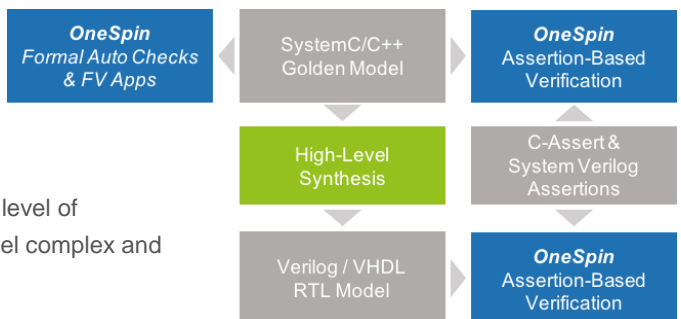
# OneSpin® 360 DV-SC™

## SystemC/C++ Extension to OneSpin 360 DV™

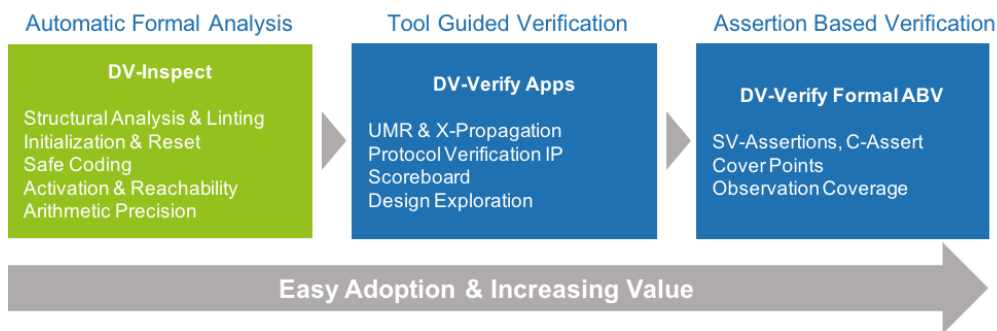
### FORMAL VERIFICATION OF ALGORITHMIC AND HIGH LEVEL SYNTHESIS MODELS

High-Level Synthesis (HLS) transforms algorithmic and potentially untimed design models often written in SystemC and C++ to fully timed Register Transfer Level (RTL) design blocks. The primary verification requirement is to allow thorough verification of algorithmic code prior to synthesis, in order to ensure that the abstract algorithm implementation is tested and fully optimized against the original specification, as well as avoiding long debug cycles.

Simulation-style verification of HLS code is largely performed by compiling and debugging the design representation, linked with the a SystemC Class Library implementation, in a similar fashion to software test. Due to the limited availability of SystemC verification tools, much of the verification task is performed on the resulting synthesized RTL code, introducing a level of indirection that makes correcting issues at the SystemC/C++ level complex and time consuming.



**OneSpin's SystemC/C++ Extension to OneSpin 360 DV enables formal verification for easy bug detection and functional verification of SystemC/C++ code.**



Coding issues, such as use of undefined values from uninitialized memory reads, or undefined operation results (e.g. division by zero), race conditions between threads, as well as SystemC-specific automated arithmetic overflows, and number precision problems may be

effectively detected using **OneSpin 360 DV-Inspect™** formal auto checks. Automated **DV-Verify Apps** at the SystemC/C++ level, allow thorough functional analysis, e.g. for data flow integrity. Support for C-Assert and temporal (with timing) System Verilog Assertions (SVA) as part of **DV-Verify Formal ABV** enables full assertion-based formal verification, including popular coverage metrics such as Quantify™ observation coverage. OneSpin 360 DV-SC works with major high level synthesis tools like Cadence Stratus™ including Stratus SystemC IP library (cynw\*) and NEC Cyber-Workbench™, as well as C/C++ code generated from Matlab/Simulink™. A versatile debugger and analyzer targeted at SystemC is integrated in the tool.

## ONESPIN 360 DV-SC FEATURE OVERVIEW

### SYSTEMC/C++ LANGUAGE

- Generic C/C++ with static memory layout, and static polymorphism, including pointers, fixed sized arrays, structs, classes, templates
- Standard Synthesizable SystemC, including sc\_fixed with QM\_TRM and QM\_WRAP
- Models for HLS, such as Cadence Stratus™, including Cadence Stratus™ Synthesis cynw\* libraries, and NEC Cyber-Workbench™
- Algorithmic C/C++ models without floating point types, e.g. generated from Matlab/Simulink™
- Mixed Language with Verilog/System Verilog/VHDL

### FORMAL AUTO CHECKS / SUPER LINTING

- Initialization/Reset Checks
- Arithmetic Overflow and Precision Checks for SystemC sc\_int, sc\_uint, sc\_fixed/sc\_ufixed, Cadence cynw\_fixed/cynw\_ufixed, and C++ base types (e.g. uint8\_t)
- Range and condition Checks, e.g. array access out of bounds, division by zero
- Race detection between threads, e.g. write-write races

### FORMAL COVERAGE ANALYSIS

- Dead Code detection, line/block coverage
- FSM deadlock detection, state/transition coverage

## ONESPIN SOLUTIONS

OneSpin Solutions is a pioneer of advanced, award winning formal verification technologies, incubated at Infineon and leveraging 300+ engineering years of development and applications experience. The company's product line includes automated design analysis requiring no knowledge of formal methods, to powerful, exhaustive, coverage-driven property verification, and incorporates an Equivalency Checker used as a gold standard within other design tool development programs. Excelling in ease-of-use, high-performance and accessibility, OneSpin's products have been leveraged by a large number of electronic system and semiconductor companies worldwide on many leading edge designs. The company operates globally, with its headquarters in Munich, Germany, and offices in San Jose, CA and Tokyo, Japan.

### YOUR NOTES:

---

---

- Generation of SystemC and System Verilog test benches
- Design consistency checking between SystemC/C++ and RTL

### FORMAL APPS

- Uninitialized Memory Read (UMR) Checks and X-Propagation Checks
- Interface Protocol Checks using Verification IP, e.g. ARM APB/AHB/AXI
- Data-Flow analysis using Scoreboard Verification IP
- Connectivity Verification

### FORMAL ASSERTION BASED VERIFICATION

- C-Assert, System Verilog Assertions, OneSpin TiDAL™ SVA
- Quantify™ Formal Observation Coverage

### INTEGRATED DEBUGGER & ANALYSER

- Waveform Debugger, Source Code Debugger, Design Instance / Hierarchy Browse
- Fan-in / Fan-out / Driver Tracing through source code
- Active Value and Active Code Annotation
- Native Value Annotation for SystemC and cynw data types, e.g. fixed point data type

**Contact** · [info@onespin.com](mailto:info@onespin.com) · [www.onespin.com](http://www.onespin.com)

USA: +1 408 734 1900 · Europe: +49 89 99013-0 · Japan: +81 3 4530 3865