Verification
CADT Needs
Semiconductor Research Corporation
May 2015

The Computer-Aided Design and Test thrust of the Semiconductor Research Corporation is seeking innovative research leading to significant advances in verification to benefit its member companies. This document summarizes member company needs in this area. These are not in priority order.

VERIFICATION
V1 Verification Core Technologies
V1.1 Advances in the scalability of automated model checking and sequential equivalence checking techniques for bit-level and bit-vector models
V1.2 Advances in techniques: general-purpose SAT solvers; constraint solving techniques; SAT solvers tuned for specific applications; automatic abstraction and abstraction-refinement; satisfiability modulo theories (SMT)
V1.3 Novel and improved algorithms for optimizing design and specification logic.
V2 RTL Verification
V2.1 Effective mechanisms for definition of assertions and coverage from an executable specification language. Coverage visualization methodologies and applications to verification (both digital and AMS) and emulation
V2.2 Effective methods to measure design-specification correctness against an RTL design (i.e., heuristics to determine verification quality and completeness).
V2.3 Techniques to generate multiple counter-examples from a set of failing assertions, and to repair failing assertions assuming a correct design and faulty assertion implementation.
V3 System-Level Verification including SoC
V3.1 Bridging pre-silicon to post-silicon verification; post-silicon validation planning, verification, and debug
V3.2 Verification of communication fabrics, with focus on liveness and non-functional properties such as quality-of-service
V3.3 Validation of SoC system integration, including validation of interface functionality; specification, detection, and validation of constraints arising from HW/FW/SW integration; specification and analysis of system-level flow compliance.
V3.4 Resilience techniques to assess and verify system functionality in the presence of soft or hard errors
V3.5 Use of specifications to support software development, such as RTL checkers, and to generate validation tests
V3.6 Co-verification of systems containing hardware and software/firmware components

V4 Analog/Mixed Signal
V4.1 Realistic analog / mixed signal specification, formal verification, equivalence checking, and coverage analysis
V4.2 Methods to help designers create models at various levels of abstraction, and to demonstrate the equivalence or correlation between models at various levels of abstraction. Attention should be paid to readability, maintainability, and particularly debugability of any automatically generated models.
V4.3 Machine learning and data mining applications on AMS circuit behavior and verification
V5 Security and Emerging Applications of Formal Methods
V5.1 Methods and models for hardware security, including the hardware/software interface.
V5.2 Efficient verification of SoC, platform, and system level non-functional properties, especially power, performance, error injection, and safety.
V5.3 Power/performance modeling of software and combined hardware/software systems
V5.4 Machine learning / data mining / big data analytic techniques for verification, validation, emulation, and diagnosis for functional and non-functional properties

Contributors:

Freescale	Jay Bhadra, Wen Chen, Monica Farkash, Jing Huang
GLOBALFOUNDRIES	Ali Wehbi
IBM	Jason Baumgartner, Flavio de Paula
Intel	Zurab Khasidashvili, Robert Nguyen, Sandip Ray
Mentor Graphics	Jeremy Levitt
SRC	William Joyner, David Yeh
Texas Instruments	Rajesh Berigei, Shane Stelmach, Noam Teutsch, David Thompson