Research Challenges in Computer-Aided Design and Test
June 2021

The Computer-Aided Design and Test (CADT) research program at the Semiconductor Research Corporation (SRC) is soliciting university proposals to address challenges facing the industry. These challenges have been identified by experts in SRC member companies and they have indicated that machine learning techniques applied to the challenges are of keen interest but not necessarily required in successful submissions. Researchers are encouraged to utilize industry standard tools as a basis or reference point, and to specify clearly not only how their research goes beyond the industry state-of-the-art, but also how they would put their results into practice in an industrial setting alongside existing methodologies and tools.

This document has four major sections:

- Functional Safety Tools and Techniques
- System, Logic, and Physical Design Tools (SLPD)
- Test
- Verification

SRC has also released a document called the Decadal Plan for Semiconductors (www.src.org/about/decadal-plan/) which describes five “Seismic Shifts” facing the electronics industry in the coming decade. Research should address issues arising from one of them:

- Smart Sensing – The Analog Data Deluge
- Memory & Storage – The Growth of Memory and Storage Demands
- Communication – Communication Capacity vs. Data Generation
- Security – ICT Security Challenges
- Energy Efficient – Compute Energy vs. Global Energy Production

While a particular research submission might address the challenges of multiple shifts, each investigator should choose one which best aligns to their effort, perhaps at the end application level. If aligned with these challenges and needs, applying Machine Learning (ML) and Artificial Intelligence (AI) techniques to enhance the capability and performance, are of interests. Researchers are encouraged to utilize the relevant SRC thrust program research works to go beyond the state of the art.

Moving forward, the SRC is also embarking on an effort to broaden participation in its funded research programs. This aggressive agenda will help us drive meaningful change in advanced information and communication technologies that seem impossible today. In the programs we lead, we must increase the participation of women and under-represented minorities as well as strike a balance between U.S. citizens and those from other nations, creating an inclusive atmosphere that unlocks the talents inherent in all of us. Please visit, https://www.src.org/about/broadening-participation/, for more information about the 2030 Broadening Pledge.

This needs document is driving the CADT research program solicitation. It is issued to US and international universities. The call for research may be addressed by an individual investigator or a research team. Our selection process is divided into two stages. The interested party is requested to submit a brief one page white paper. The white paper should clearly identify what can be done in three years. A selected white paper will result in an invitation to submit a full proposal. These proposals will be further down-selected for SRC research contracts.

Investigators who are funded will be expected to publish at top-tier conferences, including but not limited to ITC, DAC, ICCAD, ISSCC, VLSI, HPC, ISCA (part of Federated Computing Research Conference), and ESWEEK (CASES, CODESISSS, & EMSOFT). Also, if open-source software is to be developed, SRC encourages the use of MIT licensing terms when made available https://opensource.org/licenses/MIT.

White Papers for all the categories below will be considered for funding. Investigators are limited to participation in two white papers in this CADT solicitation. Submissions should highlight major category needs that are addressed, e.g. "F1", as well as which the seismic shift that the research would impact.

Note: the listed needs are not given in an organized priority ordering.

CONTRIBUTORS

Arm	Wade Walker
IBM	Gi-Joon Nam
Intel	Noel Menezes, Suriya Natarajan
NXP	Brian Mulvaney, Xiankun (Robert) Jin
Siemens EDA	Valeriy Sukharev
TI	Saqib Malik, Rubin Parekhji
SRC	David Yeh

Functional Safety Tools and Techniques

F1 Safe System

This subject summarizes the interests of member companies on tackling the safety of a system from its high-level modeling all the way down to the suitable fault models of devices aiming field failures. It covers the wide range of desired methods to model a complicated system, to perform safety analysis of the system, to perform fault injection to report diagnosis coverage, to enable fault simulation at different design abstraction levels and relate coverage across those levels, and to select effective in-field test and safety mechanism. Methods to approach safety analysis in a holistic and automated way which could cover different level of system hierarchy and complexities are very important in the space of functional safety.

F1.1	Fault models in the context of functional safety that address latent defects, aging, and other physical and electrical disturbances in the field, and methods to efficiently focus on identifying and addressing faults at either end of the spectrum (unimportant or safety-critical) including in the context of application-specific hardware
F1.2.	Methods to model, simulate, validate and test a safe system, especially large system with emerging (machine learning) accelerators
F1.3	Methods for safety metrics evaluation against safety goal. o Fault injection and fault simulation for evaluating coverage metrics at various levels of hierarchy o Verification of faults injected at component level with system reaction observed at high level o Methods to quantify the percent coverage from low level HW faults o Methods to map faults to higher level failure models, and quantitatively analyze system consequences of low level faults
F1.4	Methods to generate and select and deliver effective in-field test and diagnostic stimuli, including through intelligent / directed / synthesized work-loads to target the safety goals – both as applications that run natively (functional tests) and tests applied structurally (e.g. through scan DFT)

F2 Data Mining and Failure Prediction

Critical applications, e.g. automotive, industrial, medical, space, “always-on”, etc., maintain a good field failure return mechanism which allows the member companies to analyze the failures to improve quality assurance and/or to enhance design robustness. The field failures are also invaluable in obtaining deep understanding in the failure mechanisms, mission profiles, and effectiveness of the safety mechanisms. In turn, the knowledge gained through field failure analysis can be applied to predict the probability of failures over the device’s lifetime which becomes critical in autonomous vehicles and other unassisted applications. The member companies are asking researchers to come up with effective methods to analyze the field failure data through preferably machine learning techniques and apply the knowledge to enable failure prediction over device’s lifetime. Automated methods to perform real-time analysis of in-field degradation to detect and communicate imminent failure is needed. Getting access to the data will be one of the most daunting tasks for the researchers and creating an effective collaboration model with convincing plans and/or demonstrated history is highly valued.

F2.1	Methods assisted by machine learning to collect and analyze field data for safety analysis

F2.2	Methods of failure prediction assisted with aging and reliability model as well as system history
F2.3	Methods to design and/or select hardware and software signals to monitor during system run-time to enable reliable data collection and real-time analysis for predicting impending failure
F3 Design for Functional Safety	
To ensure system safety, a top-down safety analysis from OEM to IP level is the state-of-the-art method to drive system architecture definition and safety mechanism addition/removal. The IPs available today often cannot provide the adequate information, e.g. diagnostic coverage or built-in safety mechanisms, that is needed to reach system-level safety goal. Designing IPs with architectures/topologies as safety element out of context and/or designing infrastructure to allow inter-chip on-line testing and diagnosis are of great interest of member companies. Automated methods can assess and comprehend component vulnerabilities to help design and synthesize robust systems.	
F3.1	Design automation of architectures of digital, memory, clock, power management and data converters designed to enable those IPs as Safety Element out of Context (SEooC)
F3.2	Design automation of inter-chip infrastructure to enable safety mechanisms to cover I/O, pad-ring, packaging failures
F3.3	Design automation to help design self-checking functional/structural/application-workload-specific test patterns to achieve sufficient diagnostic coverage
F3.4	Design of hardware assisted closed-loop / on-chip methods for application re-calibration and tuning for functional safety
System, Logic, and Physical Design Tools	
S1 System Tools: Key Design Goals – Power Efficient High-Performance Designs, Long Term Reliability	
S1.1	Tools for system-level analysis and design. This includes, but not limited to: • Techniques for systems that functions across broad range of performance • Tools and methodologies to address reliability and robustness for power efficient high-performance designs • Techniques to evaluate and implement adaptive self-test design methods for use at the time of manufacture as well as in-field, to aid in diagnosis, isolation, and repair • Tools and methodologies to integrate homogeneous/heterogeneous design blocks for SoC designs

S1.2	Planning, exploration, and design tools for homogeneous/heterogeneous multi-core systems, including innovative and efficient communication fabrics, clock distribution, etc.
S1.3	Logic/physical/high-level synthesis and cross-boundary optimization. Cross-level optimization tools that can propagate physical implementation details up to the system level. This topic includes, but is not limited to • 3D design flow tools and trade-off analyses including synthesis, floor-planning, placement, power delivery, clock-tree synthesis, routing, DFT, thermal analysis, etc. • Predicting the electrical performance impact of mechanical stress • Tools providing visibility of the thermal gradients across the entire 3D Fabric and helping to avoid overdesigning and to optimize for Power-Performance-Area (PPA) thru-out the system • Tools for accounting the localized heating effects. Thermal cycle induced from self-heating is more of a concern than constant high temperature. Fatigue induced voids in signal lines could be confused with the EM induced voids. Different physical origins should require different design actions to be done to avoid this threat. • Tools for modeling the interaction between the circuit performance/reliability and the environment, for example the effect of IP placement in the layout space on the performance/reliability (what-if analysis).
S1.4	Tools and methodologies to facilitate technology evaluation, comparison and assessment at design and/or system levels for effective design space exploration that can lead to higher quality of designs. Examples include process migration of a given design, library migration to a different process, and targeting multiple processes with a single design
S2 Tools for Design Robustness	
S2.1	Tools to reduce the design margin from timing and reliability guard bands between sign-off and silicon. Tools and optimizations for accurate performance, power, and reliability margins of analog, digital and mixed-signal designs at functional and physical levels for automotive (high reliability & high yield) and low power designs
S2.2	Tools and optimizations to reduce infant mortality, to comprehend in-field degradation mechanisms like aging to achieve graceful failure, or to improve reliability against aging effects
S2.3	Tools for designing a chip with a required functionality and performance while satisfying the intended lifetime of the product
S3 Analog Tools	
S3.1	Tools for analog DFM that focus on critical devices (matched pairs, current mirrors, etc.) and minimize unaccounted exposure to random and systematic variations (e.g., layout dependent effects)
S3.2	Tools for modeling and automation of design of multiple passives (MEMS, inductors, capacitors) as well as sensors
S3.3	Advanced simulation tools for analog design including sensitivity to latent and hard defects or aging effects
S3.4	Analog synthesis and optimization, taking into account designer intent
S3.5	Tools and methodologies for smart sensing systems

Test

T1 Test of Machine Learning Systems

T1.1	HW and SW methods to evaluate correctness of on-line and off-line learning to ensure continued acceptable performance for learning/prediction during real-time application in the field
T1.2	Data sets for test and validation of learning system before deployment
T1.3	Metrics for correctness and acceptably correct inference

T2 Test Cost, Quality, and Yield Improvement

T2.1	Reducing the cost of test through various means including adaptive test such as customized test per die and burn-in elimination through pre-package stress
T2.2	Novel design-for-defect-tolerance methods
T2.3	Methods and metrics to expose and isolate subtle defects and marginalities, (e.g. those that cause Vmin degradation), in digital and mixed-signal systems
T2.4	Methods to ensure reliable acceleration of partially formed latent defects prior to screening, exploring levers that include targeted stress stimuli, e.g. elevated voltage, elevated temperature and stress duration
T2.5	Composite fault models and associated test content generation for efficient defect detection
T2.6	Approximate test methods accounting for end-user application, design and process robustness, and measurement variability

T3 High-Level Test, Validation, Diagnosis and Repair

T3.1	System-level test and in-system debug
T3.2	Bridging pre-silicon to post-silicon verification
T3.3	Methods to bridge test stimuli from functional and structural (scan) methods

T4 Analog, Mixed-Signal, RF, and High-Speed Test

T4.1	DFT methods, including BIST and BERT (bit error rate test), with coverage and test metrics to detect defects/marginalities in analog, high speed, I/O, sensors, and RF circuits and systems
T4.2	Models and methods to generate test stimuli to address coverage of gross defects considering critical-area-based analysis and practically relevant models for parametric excursions in analog/mixed-signal circuits

Verification

V1 Verification of Machine-Learning Systems

V1.1	Formal and semiformal approaches to verification of machine learning systems including defining metrics, establishing confidence, and showing transparency in decisions making
V1.2	Approaches to bound behavior for ML systems, especially those that are adaptive in nature

V2 Machine Learning Techniques for Verification	
V2.1	Learning algorithms for specification and verification of digital and AMS systems
V2.2	Learning algorithms targeted at efficient functional verification, debug, triage, and coverage closure
V2.3	Learning algorithms for better post-Si debug
V3 System-Level Verification	
V3.1	Formal and semi-formal methods applied to security and emerging applications
V3.2	Scalability of dynamic verification techniques
V3.3	Techniques to assess and verify system functionality/resilience in the presence of soft or hard errors
V3.4	Efficient verification of SoC, platform, and system level non-functional properties, especially power, performance, security, and safety
V3.5	Co-verification of systems containing hardware and software/firmware components
V4 Formal Technologies for Specification, Design, and Verification	
V4.1	Improved analog/mixed-signal specification, verification, equivalence checking, regressions, and coverage analysis
V4.2	Improved scalability and functionality of core technologies including automated model checking, solvers, synthesis for verification, and security verification
V4.3	Formal techniques for comprehensive component interface and system integration functional specifications
V4.4	Agile/incremental verification techniques that tightly integrate with agile/incremental design
V4.5	Techniques to enable provably correct system integration or correct-by-construction