Research Needs: Computer-Aided Design and Test (CADT)

May 12, 2022

Semiconductor Research Corporation (SRC), Durham, NC 27703

The Computer-Aided Design and Test (CADT) research program at the Semiconductor Research Corporation (SRC) is soliciting proposals from universities to address challenges facing the industry. These challenges have been developed by SRC member companies. To address these challenges, Machine Learning techniques are of keen interest but not necessarily required in successful submissions. Researchers are encouraged to use industry standard tools as a reference point to highlight how their research goes beyond the industry state of the art and the way to put their results into practice in industrial settings along with existing methodologies and tools.

This document has five major sections:

- Functional Safety Tools and Techniques
- System, Logic, and Physical Design Tools (SLPD)
- Test, Yield, and Post-Silicon Validation
- Verification
- New Frontier for Scalable, Correctness-Assured Hardware Design

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. The submitted research should address issues from one or more from them:

- Smart Sensing – The Analog Data Deluge
- Memory & Storage – The Growth of Memory and Storage Demands
- Communication – Communication Capacity vs. Data Generation
- Security – Information and Communications Technology (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 (i.e.at the end application level)

Moving forward, 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 used to drive CADT research program solicitation for the universities from US and its allies. 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 1-page white paper. The white paper should clearly identify in the research scope, expected outcome and deliverables 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.

Research Needs: Computer-Aided Design and Test (CADT)

May 12, 2022

Semiconductor Research Corporation (SRC), Durham, NC 27703

Investigators who are funded will be expected to publish their research results 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.

Five categories, listed below, are considered for this CADT white paper solicitation. Each investigator is limited to participate in two white paper submissions. Submissions should highlight their research category in this research need document, such as "F1" as well as which seismic shift that the proposed research would address.

CONTRIBUTORS

Arm	Wade Walker
IBM	Gi-Joon Nam
Intel	Noel Menezes, Suriya Natarajan, Jin Yang
NXP	Xiankun (Robert) Jin
Siemens EDA	Jeremy Levitt, Valeriy Sukharev, Dusan Petranovic
Texas Instruments	Saqib Malik

Note: the following needs are not listed in an organized priority order.

Functional Safety Tools and Techniques

F1 Safe System

This subject summarizes the interests from member companies to tackle 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 – both as applications that run natively (functional tests) and tests applied structurally (eg. through scan DFT)
F2 Data Mining and Failure Prediction	
Filed failure return mechanism is essential to industry members, including the companies from automotive and aerospace industries, to analyze the failures to improve quality assurance and/or to enhance design robustness. The field failure is also important to obtain clear 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 failure and the device lifetime--which is critical in autonomous driving. This research call seeks for innovative methods from researchers to analyze the field failure data through preferably machine learning techniques and apply the knowledge to enable failure prediction over device lifetime. Automated methods to perform real-time analysis of in-field degradation to detect and communicate imminent failure is needed. Since getting access to the data will be one of the most daunting tasks for the researchers, creating an effective collaboration model with convincing plans and/or demonstrated history is highly encouraged.	
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 required to drive system architecture definition and safety mechanism addition/removal. The current IP collateral files available today often fail to provide the adequate information to ensure functional safety, including diagnostic coverage or built-in safety mechanisms that are needed to reach system-level safety goal. IP with architectures/topologies as safety element out of context and/or designing infrastructures will allow inter-chip on-line testing and diagnosis and greatly benefit industry member companies. Automated methods which can assess and comprehend component vulnerabilities to design and synthesize robust systems are highly encouraged.	
F3.1	The 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	The design automation of inter-chip infrastructure to enable safety mechanisms to cover I/O, pad-ring, packaging failures
F3.3	The design automation to help design self-checking functional test patterns to achieve sufficient diagnostic coverage

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
S2.4	Automated and low overhead (RTL2GDS) design methods to achieve robustness against silent data errors due to manufacturing defects and design/process marginalities.
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, Yield, and Post-Silicon Validation

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 data-driven methods including adaptive test such as customized test per die and defect acceleration on bare-die
T2.2	Novel design-for-defect-tolerance methods to improve yield and bin-split
T2.3	Methods and metrics to expose and isolate subtle defects and marginalities (e.g. those that cause Vmin degradation) in digital logic 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, elevated voltage, elevated temperature and stress time
T2.5	Methods to generate targeted functional and structural tests to expose silent data errors due to manufacturing defects and design/process marginalities.
T2.6	Design and ATPG methods to achieve orders of magnitude speed improvement in scan test generation with similar fault coverage and advanced fault models
T3 High-Level Test, Validation, Diagnosis and Repair	
T3.1	System-level test and in-system debug (need to expand)
T3.2	Bridging pre-silicon to post-silicon verification (need to expand)
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 BIT, 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 and 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

New Frontier for Scalable, Correctness-Assured Hardware Design

New frontier CADT research for scalable, correctness-assured hardware design is listed here. The goal of this research category is to drive the hardware design innovation through disruptive innovations including but not limited to: raising level of abstraction in hardware design, implementation through Trusted complier/behavior synthesis transformation, provably correct design construction, verification as an integral part of design evolution and others.

N1 Raise Level of Abstraction in Hardware Design as A Continuation of Software Development	
N1.1	Unified semantic framework and language treatment for software and hardware development
N1.2	Seamless software/hardware tradeoff and co-design with the same design environment
N1.3	Always-alive software + hardware model to run workloads and applications
N1.4	Formal techniques for introducing computing and data concurrency into a design
N2 Design Implementation through Trusted Compiler/Behavioral Synthesis Transformations	

N2.1	Language and framework for constructing reusable and composable design transformations, amenable to formal reasoning, for designers and by designers
N2.2	Formal techniques for full verification of transformations and for ensuring correctness in composition of transformations
N2.3	Formal techniques to enable separation-of-concerns in design, including data abstraction and refinement, functional abstraction and refinement, and algorithmic optimization etc.
N3 Provably Correct Design Construction	
N3.1	Solutions for constructing a system from components to ensure that the functional correctness of each component guarantees the functional correctness of the system
N3.2	Formal techniques for specifying and verifying the functional correctness of components
N3.3	Formal techniques for specifying the functional correctness of a system and ensuring the correctness of its construction from components by verifying individual integration steps
N4 Verification as an Integral Part of Design Evolution	
N4.1	Techniques for capturing and verifying design evolution steps to avoid end-to-end verification complexity
N4.2	Techniques to enable functionally correct design exploration with trusted components and sound composition rules
N4.3	Verification accessible to designers (understandable and controllable)