Dependable On-Chip Infrastructure for Dependable MPSOCs

Kochte, Michael A.; Wunderlich, Hans-Joachim

Proceedings of the 17th IEEE Latin American Test Symposium (LATS'16) Foz do Iguaçu, Brazil, 6-8 April 2016

doi: http://dx.doi.org/10.1109/LATW.2016.7483366

Abstract: Today's MPSOCs employ complex on-chip infrastructure and instrumentation for efficient test, debug, diagnosis, and post-silicon validation, reliability management and maintenance in the field, or monitoring and calibration during operation. To enable flexible and efficient access to such instrumentation, reconfigurable scan networks (RSNs) as recently standardized by IEEE Std 1687 can be used. Given the importance of infrastructure for the dependability of the whole MPSOC, however, the RSN itself must be highly dependable. This paper addresses dependability issues of RSNs including verification, test, and security, and their importance for dependable MPSOCs. First research results are summarized, and open questions for future work are highlighted.

Preprint

Dependable On-Chip Infrastructure for Dependable MPSOCs

Michael A. Kochte, Hans-Joachim Wunderlich
email: kochte@iti.uni-stuttgart.de, wu@informatik.uni-stuttgart.de

ITI, University of Stuttgart
Pfaffenwaldring 47, D-70569, Stuttgart, Germany

***Abstract*—Today's MPSOCs employ complex on-chip infrastructure and instrumentation for efficient test, debug, diagnosis, and post-silicon validation, reliability management and maintenance in the field, or monitoring and calibration during operation. To enable flexible and efficient access to such instrumentation, reconfigurable scan networks (RSNs) as recently standardized by IEEE Std 1687 can be used. Given the importance of infrastructure for the dependability of the whole MPSOC, however, the RSN itself must be highly dependable.**

This paper addresses dependability issues of RSNs including verification, test, and security, and their importance for dependable MPSOCs. First research results are summarized, and open questions for future work are highlighted.

***Keywords*-Dependability, on-chip infrastructure, reconfigurable scan network, IEEE Std 1687, iJTAG, hardware security**

I. Introduction

Complex homogeneous and heterogeneous Multiprocessor SOCs (MPSOCs) incorporate a large amount of specialized infrastructure. This on-chip infrastructure facilitates efficient realization of increasing dependability requirements on the system [1], as for instance high availability and reliability, easy reparability and maintainability in the field, or security.

Infrastructure in current and future MPSOCs comprises conventional design-for-test, -diagnosis, and -yield as well as instrumentation for post-silicon validation and debug, online monitoring and calibration, and in-field system maintenance.

Diagnostic instrumentation is mandatory in manufacturing diagnosis, post-silicon validation, debug, bring-up, and in-field system diagnosis. Examples of instruments for efficient localization of silicon defects and design bugs include trace buffers, performance monitors, event counters, event triggers, or conventional scan chains [2, 3]. For complex MPSOCs, this debug infrastructure may contain reconfigurable and distributed structures across multiple clock domains [2, 4]. Typically, a centralized debug unit or processor controls the debug structures and aggregates runtime data. For systems with hundreds of cores, scalable access with both low hardware cost and low access latency is mandatory.

Test instrumentation includes test controllers, test wrappers, scan chains and structures for pattern decompression and compaction. Such instruments are used both in manufacturing test as well as for in-field test [5]. They can be controlled from external ports, such as external test equipment, accessed from other cores in the system [6], or from autonomous built-in test controllers in centralized or distributed fashion [7, 8].

Maintenance instrumentation is mainly used in regular system operation for monitoring, error detection, and reliability management. It includes, for instance, error monitors, memory repair controllers, and structures for system reprogramming and reconfiguration [9]. The captured data can be aggregated hierarchically per core or cluster of cores [10] and evaluated for instance by firmware or a dedicated monitoring process [11].

Traditionally, on-chip instrumentation is accessed via bus- or scan-based approaches. Bus-based approaches are often proprietary solutions tailored for the target system, e.g. for test purposes. In contrast, standardized scan networks constitute a low-cost, general purpose solution that is widely used for access to on-chip instrumentation [3, 12, 13]. The interface of the majority of scan-based access solutions follows the IEEE Std 1149.1 (Joint Test Action Group, JTAG).

For scan-based access, an instrument is equipped with a shift register allowing serial access via *scan-in* and *scan-out* terminals. Multiple instruments are connected to the JTAG circuitry either as separate "Data Registers", or in series, forming a single register.

Reconfigurable Scan Networks (RSNs) are more advanced scan architectures, where the address signals of multiplexers and other control signals are generated internally in the network. The values of these control signals determines the path through which the data is shifted, called *active scan path*. Such a structure can be viewed as a JTAG "Data Register" with variable length. RSNs allow highly

flexible, low-latency, and low-cost access to on-chip instrumentation [13, 14].

IEEE Std 1149.1-2013 (JTAG-2013) allows RSNs with *excludable* and *selectable* scan registers for efficient access to systems with power-gated components [15]. The recently ratified IEEE Std. 1687 (*Standard for Access and Control of Instrumentation Embedded within a Semiconductor Device*) [16], also known as Internal JTAG (iJTAG), targets advanced scan-based interfacing, integration, and access to *arbitrary* on-chip instrumentation.

Due to its increasing complexity, on-chip infrastructure itself becomes a dependability bottleneck of the system. For instance, a recent industrial report estimates that 20% of functional bugs are found in the debug infrastructure itself [17]. Since scan-based infrastructure access mechanisms may already amount to up to 30% of the chip area or 50% of the transistors [18, 19], a large fraction of defects in these structures causes test failures and reduces yield. In addition, infrastructure must remain accessible in the field, for instance for debug and maintenance. Without consideration of security, this opens side attack channels exposing system internals or allowing for system manipulation, jeopardizing system safety.

This paper discusses the challenges of dependable RSN-based infrastructure access and the recent progress of published EDA research. The next section gives a very brief overview of RSNs. Then, the challenges of design verification, access pattern generation, test of RSNs, and infrastructure security are addressed.

II. Reconfigurable Scan Networks

Reconfigurable scan networks (RSNs) are usually accessed through a JTAG Test Access Port (TAP). Compared to a static JTAG data register, an RSN behaves like a data register of *variable length*. The logic state of the RSN determines which scan registers in the RSN are currently accessible. The state can be changed by rewriting the content of accessible registers.

The building blocks of an RSN include scan multiplexers, scan segments or registers, and combinational logic to generate complex control signals. Scan registers may contain a shadow register for bidirectional communication or to generate control signals in the RSN, as shown in Figure 1. The operation of a scan register is controlled by control signals as explained below. According to IEEE Std 1687, multiplexed data paths and related control signals can be specified in addition.

An example of an RSN is shown in Figure 1. The one-bit scan registers S1 and S3 control the access to two multi-bit scan registers S2 and S4, respectively. For instance, the scan-in data is shifted through register S2 only if the previous access set S1 to 1. Control signals can also depend on external inputs and combinational logic, as shown for the second multiplexer. The path through which the scan-in data is shifted is termed the *active scan path*.

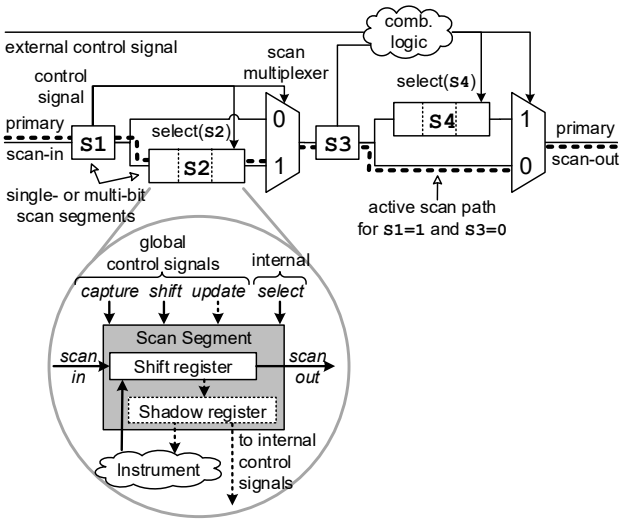

Figure 1. Example of a reconfigurable scan network and its terminology

The basic access to the RSN is an atomic operation consisting of the Capture, Shift, and Update phases (CSU), managed by the JTAG TAP controller and the corresponding control signals of scan registers. During *capture*, the scan register on the active scan path may latch new data. This data is shifted out during the *shift* phase, while new data is shifted in. Finally, during the *update* phase, the shifted-in data is latched in the *shadow latches* of the scan registers on the active scan path, which, in turn, may change the active scan path.

The state of all scan registers in the RSN is referred to as *scan configuration*. A scan configuration is valid if and only if an active scan path is formed and all elements that are not part of it are passive.

A read or write access to a scan register in the network requires that the accessed register is part of the active scan path in the current scan configuration. A *scan access* is a sequence of CSU operations, which reconfigure the RSN such that the target register(s) become part of the active scan path and which read or write the target register(s).

IEEE Std 1687 allows both hierarchical structures based on bypassing scan registers with Segment Insertion Bits (SIBs) as well as highly flexible and irregular RSNs with arbitrary control signals and distributed configuration. This can result in combinational and deep sequential dependencies between accesses [20, 21], which poses serious challenges to tasks such as verification or access pattern generation, as described below.

III. Verification

Incompatibilities of IP cores, complex dependencies in RSNs, or low level engineering changes may introduce

design bugs in the access infrastructure that limit the accessibility (increasing access times or disabling access completely). Since on-chip infrastructure is crucial for rapid production ramp-up and high product quality, thorough verification of functional correctness is mandatory to avoid costly design bugs.

For non-reconfigurable scan chains, a simple structural analysis is sufficient to check correct connectivity and timing. Additional properties such as resetability can be proven by symbolic simulation [22]. Yet these methods are unfit for general RSNs, for which sequential reachability and general model checking based approaches are required.

Proving a simple property such as the accessibility of a scan register in an RSN is in general already an NP-hard problem (equivalent to sequential ATPG) [20, 23, 14]. On the other hand, general purpose model checkers are not robust enough to handle such designs because of the high number of sequential elements and deep sequential dependencies [20].

A domain specific temporal abstraction from cycle accuracy to atomic CSU operations, called CSU-Accurate Model (CAM), has been proposed in [20, 21]. The CAM is extracted from the structural description of the RSN, for instance at register transfer level, and models the state elements, structure, and functional characteristics of RSNs in a formal way. Clauses model the constraints for RSN elements and control signals to enforce *valid* scan configurations such that an active scan path is formed along which data can be shifted.

The transition relation of the CAM defines a state transition from one scan configuration to another resulting from the application of one CSU operation. This transition relation abstracts from single capture, shift, or update cycles (as shown in Figure 2), which allows for the first time verification and pattern generation for general RSNs according to IEEE Std 1687.

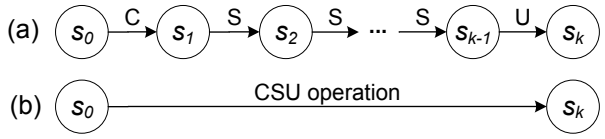

Figure 2. Abstraction of state transitions from (a) a cycle accurate model into (b) a single one in the CSU-accurate model (CAM)

In principle, the CAM can be used in any formal verification method that models the behavior with a transition relation, e.g. in symbolic or SAT-based model checkers. The use in bounded model checking has been demonstrated for proving instrument accessibility in general and under different faults in the RSN. The use in induction based reasoning allows to verify *RSN robustness*, i.e. the property that only valid scan configurations can be reached from the initial state [21].

Applying these efficient techniques allows to analyze instrument access scenarios and to discover design bugs in the access infrastructure of complex MPSOCs early in the design process.

IV. Access Pattern Generation

Finding a scan access to one or multiple target scan registers in an RSN (also called pattern retargeting) requires the computation of CSU operations such that the target registers become part of the active scan path.

For a subset of hierarchical (SIB/bypass-based) IEEE Std 1687 designs, pattern generation is a simple search that does not require backtracking and can thus be performed very efficiently [13]. For general IEEE Std 1687 designs, however, the problem becomes NP-hard. Here, the previously discussed CAM model can be applied [20].

To reduce instrument access latency, the number of cycles for accesses should be as small as possible. This can be achieved by mapping the search for a scan access to a pseudo-Boolean optimization problem, where the cost function is the number of cycles of the access [14]. This method can also be used to optimize the simultaneous access to multiple target registers (called *merged* scan accesses), for instance if a core needs to control or read multiple instruments.

Access time optimization can reduce the latency by up to 88x compared to unoptimized scan accesses in RSNs [14] and is thus crucial for efficient instrument access. The optimization is also beneficial to reduce the memory requirements for storing access patterns either on external test equipment (ATE) or on chip for online access.

V. Test of RSNs

After manufacturing of a SOC, its infrastructure, including the scan-based access mechanism, has to be tested first to ensure that test and diagnosis features are accessible and functional.

For conventional static scan chains, it has been sufficient to shift special bit sequences to detect faults in sequential elements and their interconnect [24]. In reconfigurable networks, however, complex control logic and shadow elements also need to be exercised. For the JTAG TAP controller, functional tests have been proposed to exercise the functionality of the state machine [25]. However, to completely test an RSN functionally, an exponentially high number of accesses may be required to cover all possible combinations of control signals in the RSN. Targeting a structural fault model is beneficial, since the number of single point faults only grows linearly with the size of the RSN.

A fault in an RSN may affect the sequential elements and their shadow registers, scan multiplexers, control logic, or the interconnect. The fault effects may be observable only for a certain RSN state, i.e., for a certain active scan path. For such faults, the generation of test patterns in a gate-level RSN model may require a justification over a very high number of cycles, which is beyond the capabilities of existing tools for sequential test pattern generation.

In [26], the first systematic evaluation of test methods for IEEE Std 1687 RSNs has been conducted. The fault coverage of functional and structure-oriented test sets is evaluated w.r.t. stuck-at faults at gate-level. A commercial tool for sequential test generation is used to generate patterns for the synthesized RSN. For smaller RSNs, high coverage could be reached since only few cycles need to be considered. However, the average coverage over 32 benchmark scan networks reaches only 51%.

Two functional heuristics have been developed to investigate the limits of functional test. Heuristic (F1) puts each scan segment in the RSN on the active scan path at least once and applies a flush bit sequence (e.g. ”001100”). Heuristic (F2) performs write and read operations with opposite values to each scan segment to exercise its update and capture circuitry. A minimum number of merged scan accesses (test patterns) is efficiently generated using the algorithm of [14] to reduce test time. The average fault coverage of heuristic (F1) is already 68% and increases to 77% for (F2).

To address the remaining faults, a third heuristic is employed that aims to generate a scan access that propagates the fault effect such that the active scan path in the faulty circuit is broken, altered, or changed in length. In this way, shifting is disrupted or different data is shifted in the faulty circuit, and the fault becomes observable at the primary scan output. The required constraints for test generation are expressed as Boolean formula and processed by the used SAT-based generation tool of [14]. The combination of this heuristic with (F2) results in an average fault coverage of 88% at only 5% of the runtime of the commercial tool.

To further increase fault coverage, it may be required to add dedicated design-for-test structures to the RSN. For instance, scan registers that drive control signals in the RSN can be included into a separate scan chain, or test points can be added to break sequential dependencies in the RSN. This can increase controllability and observability and reduce test generation complexity.

Recently, a functional test method for a subset of simple IEEE Std 1687 scan networks has been proposed in [27]. However, the actually achieved (structural) fault coverage is not evaluated.

Additional research is required to maximize fault coverage and to generate small test sets for static and delay fault models. The diagnosis of faults in the RSN and the computation of scan accesses in presence of faults are highly relevant problems for post-silicon validation of MPSOCs and field returns. Even partial access to a faulty MPSOC infrastructure can yield valuable insight into root causes and help to accelerate the debug or diagnosis process.

VI. Hardware Security

The increased observability and controllability of on-chip infrastructure for test, diagnosis/debug, or maintenance opens side-channels for attacks via external interfaces as well as from cores within an MPSOC. This can result in IP theft or system manipulation and safety threats. The (un)intended activation of test or debug features during operation may violate safety requirements of the system.

Still, certain levels of infrastructure accessibility are required during different phases of the MPSOC lifecycle. For example, full access is required during manufacturing test and debug, while limited access may be sufficient during operation for a power-on self test, and yet a different level of access privileges may exist for instrument access during operation by functional cores, for instance for performance monitoring, power management, or circuit calibration.

The need to secure scan-based infrastructure access has been broadly recognized [28, 29], and for conventional scan architectures, many effective techniques have been proposed to increase security [30]. Recently, the need for secure design practices has been emphasized in the context of IEEE Std 1687 [31].

To permanently restrict access to on-chip infrastructure, the access can be completely or selectively disabled by fuses, for instance after manufacturing test. Yet this becomes impractical if fine-grained access management is required. This is solved by the architecture proposed in [32, 33], where the JTAG TAP of an RSN is extended by an access sequence filter (cf. Figure 3) to inhibit accesses to a subset of protected instruments. Accesses to unprotected instruments are logically isolated from protected instruments and can be pre-computed and optimized using the method of [14]. Only these pre-computed accesses to the RSN can be applied via the respective TAP and filter. The filter monitors the control and shift data signals to the RSN and checks if the applied operations belong to the set of allowed accesses. If an illegal operation is performed, the filter goes into a trap state and inhibits any update operations. This approach is highly suitable for core-based designs since it requires no additional global signals and no modification of the TAP or RSN itself.

Multiple filters can be integrated into the access mechanism and either statically (by fuses) or dynamically (by authentication) activated. Authentication allows distinct access privileges for different entities. In simpler schemes,

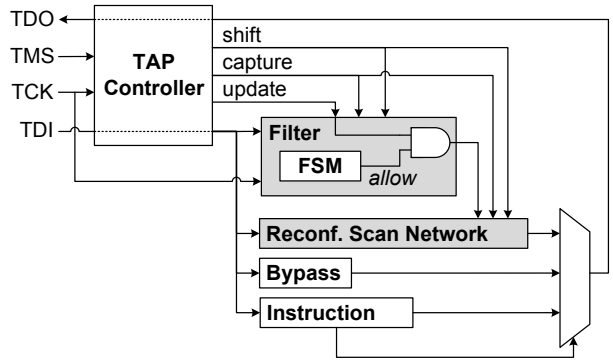

Figure 3. Filter-based RSN access restriction at the TAP [32]

each entity is assigned a secret (e.g., a string of bits), the possession of which must be proven to the chip to unlock respective instruments.

Basic authentication schemes require to present a static secret (password) to the chip to access a protected instrument. The architecture for hierarchical RSNs in [34] allows to lock and unlock access to protected instruments by replacing the enclosing bypass SIBs with *locking SIBs* (LSIBs). An LSIB is unlocked and allows to access the protected element(s) after shifting the secret key into certain positions of the scan registers in the RSN. Traps in the RSN shall impede brute force attacks. Since the secret is distributed to all authorized entities and transferred to the chip in plaintext, the probability that such schemes are eventually compromised by secret leakage is usually too large for systems with high security requirements.

Stronger authentication schemes do not reveal the secret during communication with the chip by use of challenge-response protocols. This has been employed in [35], where a hierarchical RSN is extended by an authorization instrument, which controls *secure SIBs* (S^2IBs) such that only authorized entities can access protected elements. Each accessing entity can be assigned distinct permissions. This method requires only a small modification of the original RSN and no additional global wiring for security control. Since the S^2IBs are only slightly larger than regular SIBs, this protection scheme scales very well with the number of protected instruments in an MPSOC.

While different architectures for access protection in RSNs have been proposed, proving their correctness at different stages of the design process is mandatory to find design bugs and to analyze the resulting protection level. In [36], the first unbounded model checking method for RSNs has been presented, able to efficiently handle even large design instances. The method employs the temporal abstraction discussed in Section III and implements unbounded model checking using Craig interpolation. It is used to prove or refute access protection properties in two different architectures and to analyze the impact of design mutations, modeling either design bugs or fault injection based attack scenarios.

More research is required to develop further scalable modeling approaches for the verification of complex infrastructures, allowing to reason not only about attacks at register-transfer level, but also at gate or even lower level.

VII. Conclusion

Dependable MPSOCs require a large and increasing amount of instrumentation to facilitate efficient bring-up, test, debug and diagnosis, or maintenance in the field. Reconfigurable scan networks have been proposed as scalable access mechanism to such infrastructure and recently been standardized. To ensure dependable operation of an MPSOC, the access mechanism itself must provide reliable and secure access to the instruments.

Yet the complexity and high sequential depth of such scan networks pose challenges for established EDA algorithms. Recent research proposed first solutions for some of these challenges, addressing modeling and verification, access optimization, test, or access protection in RSNs. Open problems remain in test and diagnosis, verification, and secure and fault-tolerant design, which will stimulate future research efforts for dependable infrastructure.

Acknowledgements

This work was supported by the German Research Foundation (DFG) under grant WU 245/17-1 (ACCESS) and further work by the Baden-Württemberg Stiftung (IKT-Sicherheit, SHIVA).

References

[1] A. Avizienis, J.-C. Laprie *et al.*, "Basic concepts and taxonomy of dependable and secure computing," *IEEE Trans. Dependable and Secure Computing*, vol. 1, no. 1, pp. 11–33, Jan. 2004.

[2] M. Abramovici, "In-System Silicon Validation and Debug," *IEEE Design & Test of Computers*, vol. 25, no. 3, pp. 216–223, 2008.

[3] N. Stollon, *On-Chip Instrumentation: Design and Debug for Systems on Chip*. Springer US, 2011.

[4] A. Su, J. Kuo *et al.*, "Multi-core software/hardware co-debug platform with ARM CoreSight, on-chip test architecture and AXI/AHB bus monitor," in *Proc. International Symp. VLSI Design, Automation and Test (VLSI-DAT)*, 2011, pp. 1–6.

[5] L.-T. Wang, C. E. Stroud, and N. A. Touba, *System-on-Chip Test Architectures: Nanometer Design for Testability*. Morgan Kaufmann, 2010.

[6] A. Krstic, L. Chen *et al.*, "Embedded software-based self-test for programmable core-based designs," *IEEE Design & Test of Computers*, vol. 19, no. 4, pp. 18–27, Jul. 2002.

[7] Y. Zorian, "A distributed BIST control scheme for complex VLSI devices," in *Proc. IEEE VLSI Test Symposium (VTS)*, 1993, pp. 4–9.

[8] K. J. Lee, T. Y. Hsieh, and C. Y. Chang, "On-Chip SOC Test Platform Design Based on IEEE 1500 Standard," *IEEE Trans. on Very Large Scale Integration Systems*, vol. 18, no. 7, pp. 1134–1139, 2010.

[9] Y. Zorian, "Embedded Memory Test and Repair: Infrastructure IP for SOC Yield," in *Proc. IEEE International Test Conference (ITC)*, 2002, pp. 340–349.

[10] E. Larsson and K. Sibin, "Fault management in an IEEE P1687 (IJTAG) environment," in *Proc. IEEE Int'l Symp. Design and Diagnostics of Electronic Circuits Systems (DDECS)*, April 2012, embedded tutorial.

[11] M. Benabdenbi, F. Pecheux, and E. Faure, "On-line Test and Monitoring of Multi-Processor SoCs: A Software-based Approach," in *Proc. Latin American Test Workshop (LATW)*, 2009, pp. 1–6.

[12] J. Rearick, B. Eklow *et al.*, "IJTAG (Internal JTAG): A Step Toward a DFT Standard," in *Proc. IEEE International Test Conference (ITC)*, 2005, paper 32.4.

[13] E. Larsson and F. Ghani Zadegan, "Accessing Embedded DfT Instruments with IEEE P1687," in *Proc. IEEE Asian Test Symposium (ATS)*, 2012, pp. 71–76.

[14] R. Baranowski, M. A. Kochte, and H.-J. Wunderlich, "Scan Pattern Retargeting and Merging with Reduced Access Time," in *Proc. IEEE European Test Symposium (ETS)*, 2013, pp. 39–45.

[15] "IEEE Standard for Test Access Port and Boundary-Scan Architecture 1149.1-2013," IEEE Computer Society, 2013, Test Technology Technical Committee of the IEEE Computer Society, USA.

[16] "IEEE Std 1687-2014 – IEEE Standard for Access and Control of Instrumentation Embedded within a Semiconductor Device," Dec. 2014, IEEE Computer Society.

[17] S. Kumar, "Industrial challenges of bugs and defects," in *NSF/SRC/DFG Joint Workshop on Bugs and Defects in Electronic Systems: The Next Frontier*, Apr. 2013, Schloss Dagstuhl, Germany.

[18] R. Guo and S. Venkataraman, "A technique for fault diagnosis of defects in scan chains," in *Proc. IEEE International Test Conference (ITC)*, 2001, pp. 268–277.

[19] F. Yang, S. Chakravarty *et al.*, "On the Detectability of Scan Chain Internal Faults - An Industrial Case Study," in *Proc. IEEE VLSI Test Symposium (VTS)*, 2008, pp. 79–84.

[20] R. Baranowski, M. A. Kochte, and H.-J. Wunderlich, "Modeling, Verification and Pattern Generation for Reconfigurable Scan Networks," in *Proc. IEEE International Test Conference (ITC)*, 2012, paper 8.2.

[21] R. Baranowski, M. A. Kochte, and H.-J. Wunderlich, "Reconfigurable Scan Networks: Modeling, Verification, and Optimal Pattern Generation," *ACM Trans. Design Automation of Electronic Systems (TODAES)*, vol. 20, no. 2, pp. 30:1–30:27, Feb. 2015.

[22] H. B. Kamepalli, P. Sanjeevarao, and C.-J. Park, "Scan Chain Verification Using Symbolic Simulation," Patent, 2006, US Patent App. 7,055,118.

[23] M. Keim, T. Waayers *et al.*, "Industrial Application of IEEE P1687 for an Automotive Product," in *Proc. Euromicro Conf. on Digital System Design (DSD)*, 2013, pp. 453–461.

[24] K.-J. Lee and M. Breuer, "A Universal Test Sequence for CMOS Scan Registers," in *Proc. IEEE Custom Integrated Circuits Conference (CICC)*, 1990, pp. 28.5.1–28.5.4.

[25] A. Dahbura, M. Uyar, and C. W. Yau, "An Optimal Test Sequence for the JTAG/IEEE P1149.1 Test Access Port Controller," in *Proc. IEEE International Test Conference (ITC)*, 1989, pp. 55–62.

[26] M. Schaal, "Test of reconfigurable scan-networks," Master's thesis, Institut für Technische Informatik, Universität Stuttgart, Germany, Dec. 2013. [Online]. Available: http://elib.uni-stuttgart.de/opus/volltexte/2013/8832/pdf/DIP_3380.pdf

[27] R. Cantoro, M. Montazeri *et al.*, "On the Testability of IEEE 1687 Networks," in *Proc. IEEE Asian Test Symposium (ATS)*, 2015, pp. 1–6.

[28] D. Hely, M. L. Flottes *et al.*, "Scan Design and Secure Chip [Secure IC Testing]," in *Proc. IEEE On-Line Testing Symposium (IOLTS)*, 2004, pp. 219–224.

[29] M. Tehranipoor and C. Wang, *Introduction to Hardware Security and Trust*. Springer, 2011.

[30] J. Da Rolt, A. Das *et al.*, "Test versus Security: Past and Present," *IEEE Trans. on Emerging Topics in Computing*, vol. 2, no. 1, pp. 50–62, Mar. 2014.

[31] J. Dworak and A. Crouch, "A Call to Action: Securing IEEE 1687 and the Need for an IEEE Test Security Standard," in *IEEE VLSI Test Symp. (VTS)*, April 2015, pp. 1–4.

[32] R. Baranowski, M. A. Kochte, and H.-J. Wunderlich, "Securing Access to Reconfigurable Scan Networks," in *Proc. IEEE Asian Test Symposium (ATS)*, 2013, pp. 295–300.

[33] R. Baranowski, M. A. Kochte, and H.-J. Wunderlich, "Access Port Protection for Reconfigurable Scan Networks," *Journal of Electronic Testing (JETTA)*, vol. 30, pp. 711–723, 2014.

[34] J. Dworak, A. Crouch *et al.*, "Don't Forget to Lock your SIB: Hiding Instruments using P1687," in *Proc. IEEE International Test Conference (ITC)*, 2013, paper 6.2.

[35] R. Baranowski, M. A. Kochte, and H.-J. Wunderlich, "Fine-grained access management in reconfigurable scan networks," *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, vol. 34, no. 6, pp. 937–946, June 2015.

[36] M. A. Kochte, R. Baranowski *et al.*, "Formal verification of secure reconfigurable scan network infrastructure," in *IEEE European Test Symposium (ETS), accepted for publication*, 24-27 May 2016, Amsterdam, The Netherlands.