Security Challenges During VLSI Test

David Hély
LCIS
Grenoble Institute of Technology
Valence, France
Email: david.hely@lcis.grenoble-inp.fr

Kurt Rosenfeld
Polytechnic Institute of NYU
Brooklyn, NY
Email: kurt@isis.poly.edu

Ramesh Karri
Polytechnic Institute of NYU
Brooklyn, NY
Email: rkarri@poly.edu

***Abstract*—VLSI testing is a practical requirement, but unless proper care is taken, features that enhance testability can reduce system security. Data confidentiality and intellectual property protection can be breached through testing security breaches. In this paper we review testing security problems, focusing on the scan technique. We then present some countermeasures which have recently been published and we discuss their characteristics.**

I. Introduction

Integrated circuit testing has emerged in recent years as a new security problem. Indeed, while testability requires observability and controllability of internal states, security often requires the opposite. It has been shown that confidential data and intellectual property can be jeopardized by standard design for test (DfT) techniques. The observability provided by test structures can be used by an attacker to examine the data being processed by the chip. Similarly, the test structures can leak information about the chip design. In an attacker's hands, the controllability provided by test structures can be used for inserting malicious data into a system, bypassing validation that is done at the perimeter, or forcing the system into an insecure state.

Paradoxically, testability is also critical for the security of an IC, since a flaw that is undetected during the testing process could lead to an exploitable security flaw in the field. Designers must confront a new challenge: how to achieve high test quality without lowering the security of the circuit. At a first glance, it appears that the built-in self-test (BIST) technique is the best candidate for reducing the security risk associated with testability. BIST exposes a restricted interface to the tester. The BIST routine is started, it completes, and the tester collects the results. Ad-hoc techniques have been proposed for crypto processors, taking advantage of cryptographic properties to efficiently implement BIST techniques. Nevertheless, BIST still has some drawbacks in terms of hardware overhead, fault diagnosis, and even security. Therefore, the scan path technique is still heavily used by the test community. Securing the scan technique has been a hot topic in recent years in industry and in academia. DfT is a complex task involving many actors and requiring many iterations during the design flow in order to reach the desired level of testability. DfT security schemes must mesh well with existing design tools in order to be adopted by designers. Moreover, System-On-Chip (SoC) integrators more and more deal with third party intellectual property (IP) providers, which leads to several questions. First, how can we assume that the security of the SoC will not be degraded by the test mechanisms involved with this IP? Second, how can we efficiently integrate the secure DfT of this IP during SoC integration? Finally, from the IP provider's point of view, how can we be sure that the SoC test infrastructure will not be used to attack our intellectual property?

The first part of the paper sums up the security issues associated with VLSI testing. The second section presents some countermeasures proposed in the literature. Finally, the last section discusses the characteristics of each of the presented countermeasures.

II. Testing Security Issues

A. Test Hazards

In [1], Yang et al present a new side-channel attack against the Data Encryption Standard (DES), exploiting the observability provided by the scan chain mechanism. They show the same kind of attack against the AES (Advanced Encryption Standard). These attacks show that including scan-based test structures in a crypto circuit (even if secret key registers are excluded from the chain) is a major vulnerability for the circuit. The attack consists of unloading the scan chain at different steps of the algorithm, first to determine the structure of the implementation, and then to retrieve confidential data. Testing features can also compromise security in less spectacular ways. Intellectual property protection is a major concern in SoC design [2]. An adversary can exploit test structures to gain information about the design. Indeed, scan test features can be a very powerful tool for reverse engineering. Also, test features can exploited as the entry point for fault attacks. In the case of scan chains, the scan path can be used either to insert malicious data into the chip or to gain specific information about the design that can facilitate a subsequent attack. Also, the possibility of activating the scan chain at any time may be a viable random fault injection means. Due to the security hazards present in IC testing, the following criteria should be met by any DfT technique applied to security-critical circuits:

1) Access to testing features should be restricted to privileged user **(Protected Test Mode)**
2) Confidential data handled by the circuit must never be output **(Confidential Information Leakage)**

3) In functional mode, the circuit should never process data inserted via the testing interface **(Malicious Data Insertion)**

The first item will assure that for the specific case of crypto chips, secret data will never be leaked via the scan chain. The second item is necessary to differentiate between test mode and functional mode, so that data insertion is not possible via the scan path. The third item is a requirement related to IP protection. Most security-critical SoCs have two distinct modes: test mode and functional mode (also called *user* mode or *mission* mode). In test mode, all test-specific operations are allowed. After production test, ICs are often permanently changed (e.g., by blowing a fuse) so that they only operate in functional mode, or require a key or password to return to test mode. This has security benefits, but can make maintenance cumbersome in some circumstances.

B. The Attackers

Securing ICs is done according to a threat model which includes an attacker profile. Concerning test-based attacks, different attacker profiles may be defined:

- **Authorized Test Engineer**: A test engineer who has full access to the chip while it is in test mode.
- **In-the-field Hacker**: A hacker who attempts to activate the test features of the chip after it has been deployed.
- **IP Provider**: An external party that provides hardware design modules or embedded software modules that are included in the design of the system.

For the first category, the risk is limited. The most secure chips are produced using dedicated foundries where security is a part of the management and production process. For the second category of attackers, their question is simple: "How can I activate the test mechanism?" As we have seen earlier, most SoCs have two distinct modes: test mode and user mode. An attacker's most desired accomplishment is to merge both modes to have access to testing features while the chip is user mode. A standard SoC test architecture is composed of a test controller and a test access mechanism (TAM). The test controller is connected to the tester and propagates scan data via the TAM. In the case of scan chain techniques, hackers can directly use the test controller to access IP through the internal scan chain. Where the test controller is protected against unauthorized use, attackers can still launch direct attacks on the the scan chain via brute-force methods such as die probing. Nevertheless the likelihood of such an attack succeeding is very low since it requires highly sophisticated tools and skills. Another option for hackers is to stress the design to trigger internal faults which could activate the test mechanisms. Faults can be induced by many means (voltage, optical, electromagnetic). Two kinds of attacks are then to be considered:

1) **Protocol Attack**: the attacker tries to use test features as a test engineer would.
2) **Brute Force Attack**: the attacker tries to activate the scan feature, bypassing the protocol.
3) **Infiltration Attack**: the attacker sniffs, modifies, or injects data on the test bus from a core within the chip.

C. Design Challenges

Protecting data confidentiality and intellectual property against testing hazards should not be a new, hard-to-meet constraint for designer and test engineers. High test coverage is essential for security because production flaws could induce a system malfunction that could be a security hazard when the system is working. High test coverage is already a goal in most IC development efforts. Likewise, security certification requires quantitative assessment of test coverage during the certification process. Design for testability is already a very quantitative and well-automated process, so additional security constraints can fit in with usual DfT tools such as scan insertion and automatic test pattern generation. Moreover, most modern SoC are made of many IP modules which have to follow predefined rules so that their test interfaces can be easily integrate within the SoC. Finally, both security and economics favor simpler designs; security because simple designs are easier to verify, economics because simple designs use less die area and less power. Production test is already a costly stage of the VLSI production process. Ideally, security enhancements should not complicate the test process, increase test time, or increase the complexity of the automated test equipment. The main parameters which characterize secure DfT techniques are then:

- the test coverage, the test penalty
- the compatibility with tools, design penalty
- resistance against protocol attack
- resistance against brute attack

III. COUNTERMEASURES

Several techniques have been proposed in the literature to mitigate test-related security hazards. They address the test protocol, scan chain design, and/or test pattern generation.

A. Protocol Level

Reinforcing the protection based on the differentiation of the test mode from the user mode is an approach widely adopted by the SoC industry. The test protocol solution consists of disabling the test feature in user mode (i.e., when the system is handling confidential data). To protect confidential data processed by the circuit, such as keys, it has been proposed in [3] to modify the usual test protocol so that before entering in test mode, the circuit is completely reset. Then the reset circuit is checked. If it is correct then scan-in, capture, and scan-out operations can be done. Then before returning again to the functional mode the circuit is again reset so that no data insertion can be done using the scan path. In [4], the authors proposed to isolate the registers containing confidential data from the scan chain, and they require a global reset when switching between user mode and test mode. Both of these solutions require modification of the test protocol. In the first case, the protocol modification consists in adding the reset operation before entering test mode (i.e scan-in, capture and

scan-out). In the second method, the test controller has new signals to control a special secure register, the so-called MKR (Mirror Key Register) which guarantees that secret information will never be leaked via the scan chain. Rosenfeld [6] proposed a crypto-based scheme for protecting JTAG interfaces against protocol-level attacks like sniffing secret data on the test bus. The scheme has the added benefit of authenticating chips in a system, thus thwarting supply chain attacks where counterfeit parts are delivered to system integrators. Although the scheme uses lightweight crypto, it still has area overhead and adds some test time overhead for initializing the crypto state.

B. Scan Chain-Level Protection

We have seen that attackers can exploit their ability to activate the scan chain directly bypassing the logic controlling the switch between the user mode and the functional mode, thanks for instance to probing attack. Methods such as scan data scrambling offer some protection against such an attack. Still, by applying a brute-force attack on a scan chain part, the scrambling will lose its effectiveness, and the protection level will be determined by the length of the scrambled parts. It is then necessary to add dedicated protection to ensure that the scan path is not activated during user mode. In [4], the authors propose to add data integrity logic control to the signals driving the MKR so that they detect any fault or brute attack leading to scan activation during the user mode. In [3], the authors identify the scan-enable signal as a high-risk signal since it drives the scan function of each scanned flip-flop. They then propose to add an integrity mechanism to this signal so that any activation of this signal while the chip is in user mode will trigger a hard reset of the circuit. Another method is proposed in [7], consisting of a dedicated flip-flop inserted into the scan chain at register-transfer level. The only purpose of this flip-flop is to detect any scan chain activation during user mode. This so-called "spy flip-flop" is functionally stuck at one value. The only way to change the value is to scan in the opposite value. If the opposite value is read on its output, the detection mechanism is activated. Other countermeasures have been proposed for making scan-chain shifting not exploitable by attackers. In [8], the authors propose to scramble the scan path so that even if an attacker can dump the scan path in user mode, the data are difficult for him to interpret. In authorized test mode, the scan path is normally connected, while in user mode the scan path segments are randomly connected. In [10], Lee et al. proposed an evolution of the scrambling technique, the so called "lock and key" technique. The scan path is divided into smaller parts and the scan data are shifted into an LFSR, essentially seeding the pseudorandom sequence generator. If the prefix of the scanned-in data is the test key, the scan parts are connected in a predictable way. Otherwise, connections are not predictable. In [9], the authors present a methodology based on partial scan and logic obfuscation. The partial scan insertion method is based on standard partial scan techniques with an additional weight on sensitive registers so they are excluded from the scan path. The technique still relies on the differentiation of test mode and user mode. Such a solution does not completely prevent brute-force attacks.

C. Communication Security for the Test Channel

In an SoC, each of the cores needs to be tested. Regardless of whether BIST is used or explicit structural or functional testing is used, the test responses need to be communicated back to the tester. To conserve wiring on the die, it is common to use shared wiring (bus or daisy-chain) to connect multiple cores to an on-chip test controller. If the cores of the SoC are not completely trustworthy, the shared wiring can be a security issue. Depending on the details of how the shared wiring is implemented, the chip can be vulnerable to infiltration by a malicious core, which can sniff, inject, or modify bits in the communication channel between the test controller and a benign victim core. This threat can be mitigated by using crypto techniques on the test bus. Rosenfeld and Karri [5] propose an architecture for establishing a session key secure test communication between a test controller and the cores in an SoC. The idea is to take advantage of the fact that although the SoC integrator has little knowledge or control of circuitry inside third-party IP cores, he has full control over the intercore test wiring, the topology of which forms a root of trust.

D. Pattern Watermarking

Pattern watermarking is an efficient way to implement scan chain protection, in [11], the scan chain is always enabled, but each time data are scanned-in, the first value is compared with a golden reference in order to authorize or reject the rest of the scan operations. Paul et al [12] proposes also a method based on adding some flip-flops within the scan path, then the circuit will switch to test mode only if the appropriate sequence has been shifted in to these flip-flops.

IV. Solution Comparison

A. Security

As seen above, secure DfT techniques can be classified into four main categories: test protocol security, scan chain integrity, data scrambling, and pattern watermarking. Securing the test protocol, as seen in [1] and [3], has the advantage that even if one bypasses the authentication and uses the test features as a test engineer, no confidential data will be outputted. If data scrambling or pattern watermarking are implemented without protocol security, an attacker who gets the authentication will furnish the correct authentication to disable the scrambling or the watermarking and will then be able to unload confidential data. Concerning the overall security of a chip, test protocol security mechanisms address only the issue of data confidentiality. To avoid reverse engineering via analysis of scan data, scan data must be encoded, as is the case in user mode with scrambling protection or integrity mechanisms. However, in authorized test mode, the scan data are fully interpretable for most of the techniques. IP protection relies then on the non-disclosure agreement between the foundry and the IP owner. To address this issue, [9] proposes an improved solution using partial scan and a

Attacks	Test Protocol	Scan Chain Integrity	Data Scrambling	Pattern watermarking
Protocol Attack	++	-	-	+
Brute Attack	–	++	+	-
Secure Item	Test Protocol	Scan Chain Integrity	Data Scrambling	Pattern watermarking
IP Protection	–	+	++	-
Confidential Data Protection	++	++	+	-
Constraints	Test Protocol	Scan Chain Integrity	Data Scrambling	Pattern watermarking
Integration	++	-+	+	++
Test Efficiency Impact	++	++	++	–
Design Impact	++	-	-	+

TABLE I
SECURE DFT TECHNIQUES OVERVIEW

scrambling circuit in test mode. Partial scan makes design recovery more difficult, and scrambling even more so. In the case of brute-force attack, the security of the test protocol itself is far not enough, indeed for such an attack the protocol is bypassed and the scan directly activated by internal scan signals. Scan chain integrity mechanisms are thus a good complement to protocol security and are proposed in [4] and [3]. Data scrambling is by itself a good protection against brute attack. Indeed, even if the scan path is activated, data are difficult to analyze as depicted in [10]. In [9], the scrambling is enabled only in test mode then in case of brute attack, the data can still be exploited. Nevertheless it is to be noticed that the scrambling efficiency will rely on the segment length, the smaller are the segments the more efficient is the protection. In terms of security, there is no universal protection, and as depicted in Table I, merging at least two kinds of protection in a design is necessary to overcome both protocol and brute-force attacks.

B. Design constraints

Test protocol security is easy to integrate into the IC development process and can be done during the SoC integration without IP core modification [1], [3], [9]. In terms of test operation, there is little impact (no more test data) except that the test setup may be a longer. This fixed-time setup latency is negligible when compared to the overall test time. Scan path integrity mechanisms do not impact the test time either. However the integration effort may vary. In [3], the integrity mechanism proposed must be designed after synthesis and scan insertion at the gate level, which is not very convenient. The solution proposed by [4] or [10] is easier to implement but does only prevent from scan activation of sensitive registers. In [7], the spy flip-flop can be implemented at RTL level by the IP designer himself but adds cells to the scan path, thus increasing test time and design area. Pattern watermarking [13] can be easily integrated at RT-level using scan insertion constraints. The drawback of such a solution is the additional data to be scanned from the scan path, resulting in additional test time and test data, and therefore increasing the overall test cost. Finally, a data scrambling-based solution [8], [10] imposes to specify scan segments and to add additional circuitry to drive the mechanism.

V. CONCLUSIONS

We have surveyed the security issues associated with scan-based test in modern VLSI. The main threat mitigation techniques in the literature have been discussed, along with their strengths and weaknesses. In conclusion, there is no universal minimum-cost solution for protecting against arbitrary test security threats. Designers have to identify the security concerns they want to overcome and the price they are willing to pay before selecting the appropriate countermeasure.

ACKNOWLEDGMENT

The work of Ramesh Karri and his group is supported in part by National Science Foundation grants ECCS-0621856 and CNS-0831349.

REFERENCES

[1] B. Yang, K. Wu, and R. Karri,"Scan based side channel attack on dedicated hardware implementations of data encryption standard", Int. Test Conf., Charlotte, NC, 2004, pp. 339344.
[2] R.S. Chakraborty and S. Bhunia: "HARPOON: An Obfuscation-Based SoC Design Methodology for Hardware Protection", IEEE Trans. on CAD of Integrated Circuits and Systems (TCAD), 2009.
[3] D. Hély, F. Bancel, M.-L. Flottes, and B. Rouzeyre. "Securing scan control in crypto chips", Journal of Electronic Testing-Theory and Applications, 23(5):457464, Oct. 2007.
[4] B. Yang, K. Wu, and R. Karri. "Secure scan: A design for test architecture for crypto chips", IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 25(10):22872293, Oct. 2006.
[5] K. Rosenfeld and R. Karri, "Security-Aware SoC Test Access Mechanisms" IEEE VLSI Test Symposium, to appear in proceedings, May 2011.
[6] K. Rosenfeld and R. Karri, "Attacks and defenses for JTAG", IEEE Design and Test of Computers., Volume 27, Number 1, Jan 2010, Pages 35-47, Special Issue on Trustworthy Hardware.
[7] F. Bancel and D. Hély, "Integrated circuit comprising a test mode secured by detection of the state of control signal" US Patent 769419
[8] D. Hély, F. Bancel, M. L. Flottes, B. Rouzeyre, M. Renovell, and N. Brard, "Scan design and secure chip", IEEE Int. On-Line Testing Symp., Funchal, Portugal, 2004, pp. 219226
[9] M. Ino, T. Yoneda, M. Hasegawa, H. Fujiwara "Partial Scan Approach for Secret Information Protection", IEEE. European Test Symposium (ETS 2009),pp. 143-148
[10] J. Lee, M. Tehranipoor, C. Patel, and J. Plusquellic. "Securing designs against scan-based side-channel attacks", IEEE Transactions on Dependable and Secure Computing, 4(4):325336,Oct.-Dec. 2007.
[11] D. Hély, F. Bancel, M. L. Flottes, B. Rouzeyre "Scan Pattern Watermarking", 7th IEEE Latin American Test Workshop, Buenos Aires (2006)
[12] S. Paul, R. S. Chakraborty, and S. Bhunia. "VIm-Scan: A low overhead scan design approach for protection of secret key in scan-based secure chips", 25th IEEE VLSI Test Symposium(VTS07), pages 455460, 2007.
[13] U. Chandran and D. Zhao, "SS-KTC: A High Testability Low overhead Scan Architecture with Multi-Level Security Integration", 27th IEEE VLSI Test Symposium(VTS09), pages 321-326, 2009