Thwarting Scan-Based Attacks on Secure-ICs with On-Chip Comparison

Jean Da Rolt, Giorgio Di Natale, Marie-Lise Flottes, Bruno Rouzeyre

Thwarting Scan-Based Attacks on Secure-ICs With On-Chip Comparison

Jean Da Rolt, Giorgio Di Natale, Marie-Lise Flottes, and Bruno Rouzeyre

***Abstract*— Hardware implementation of cryptographic algorithms is subject to various attacks. It has been previously demonstrated that scan chains introduced for hardware testability open a back door to potential attacks. Here, we propose a scan-protection scheme that provides testing facilities both at production time and over the course of the circuit's life. The underlying principles to scan-in both input vectors and expected responses and to compare expected and actual responses within the circuit. Compared to regular scan tests, this technique has no impact on the quality of the test or the model-based fault diagnosis. It entails negligible area overhead and avoids the use of an authentication test mechanism.**

***Index Terms*— Design-for-testability (DfT), scan-based attack, security, testability.**

I. Introduction

Many aspects of our daily lives rely on electronic data interchange. Encryption algorithms are used to guarantee the confidentiality, integrity, and authenticity of these exchanges. These algorithms are implemented on dedicated hardware for performance optimization and to embed confidential information, which must be kept secret from unauthorized users.

Imperfect production processes of electronic devices lead to the need for manufacturing testing to sort out defective circuits from good ones, whatever be the target application. This is even more relevant for secure circuits where a physical defect could jeopardize the security of the confidential information.

However, the most common practice for testing digital devices relies on a scan-chains insertion that guarantees a high fault coverage and thus an ultimate product quality, but opens backdoors to security threats too. The "Scan attacks" described for instance in [1] and [2] utilize the access offered by scan chains' IOs for retrieving the secret key of an encryption core. These attacks rely on the possibility to observe the circuit's internal state while this state is related to the secret.

A common industrial practice to solve this security threat is to physically disconnect the scan chains after production testing by blowing the fuses located at both ends of the scan chains. However, this solution impedes the testing of those devices requiring being tested after manufacturing. In particular, the correct behavior of the secure circuits should be validated after the introduction of the secret key, which can be programmed at any time of the circuit's lifecycle. This secured information can indeed be owned by any circuit producer (e.g., designer, manufacturer, and system integrator) or user (e.g., reseller or final customer). In addition, scan disconnection stops any further analysis, e.g., diagnostic, or cannot be considered as an appropriate response to the scan attack if the connection can be reconstructed. In the literature, several solutions have thus been proposed to avoid disconnecting scan chains after manufacturing testing. However, these solutions are either expensive or not fully safe against new scan attacks.

In this brief, we describe a new design-for-testability (DfT) architecture that eliminates the need to disconnect the scan chains. This approach is based on the concept of withholding information. The test procedure consists in providing both the test vectors and expected test responses to the device-under-test (DUT) for an on-chip comparison.

Methods for the on-chip comparison of actual and expected test responses have already been explored in other contexts [3]–[6], mainly to reduce the test data volume to transfer from DUTs to test equipment. However, none of these solutions achieve the target security requirements since individual bit values stored in the scan chains can still be observed or deducted from observed data, thanks to the test circuitry.

Because testability features must not be implemented to the detriment of the security of the circuit, and vice versa, this brief also discusses test and diagnostic procedures with our DfT proposal, as well as security of the circuit with respect to attacks perpetrate on the test infrastructure.

This brief is organized as follows. Section II summarizes the most relevant design-for-testability-and-security proposals from the literature, and discusses their related drawbacks. The detailed implementation of the module in charge of the proposed test strategy is described in Section III, and related costs and impact in terms of insertion in the design flow are also presented. Section IV discusses security, testability, and diagnostic issues related to the introduction of the proposed test scheme. Finally, Section V concludes on this brief.

II. Related Works

Several countermeasures have been proposed to face the scan attacks, while allowing access to the scan chain after the manufacturing test. Two classes of solutions can be found in literature: the use of dedicated secure test wrappers, and the introduction of hidden functions to obfuscate the real contents of the scan chains.

Solutions based on the use of secure test wrappers basically implement an FSM with two states: mission mode and test mode. In mission mode, the circuit handles confidential data and the scan chain cannot be accessed (i.e., the scan enable is forced to 0). Conversely, scan facilities can be used in test mode because there is not any confidential data processed in the circuit in this mode. Implementing secure modes for testing without leakage of confidential data depends on: how is implemented the process for switching from (to) mission to (from) test mode; how confidential information is removed from the data flow when a switch to test mode is required; and finally, how to further protect data in mission mode against invasive attacks on the test infrastructure.

Switching from mission to test mode is usually implemented by resorting to an authentication protocol. For instance, the solution presented in [7] offers a security extension for IEEE 1149.1 standard where the test controller must receive a secret wrapper key to enable the test mode. More complex wrappers based on challenge-response protocols were proposed in [8] and [9]. However, a secured authentication method requires the implementation of crypto

Manuscript received November 5, 2011; revised February 24, 2013; accepted March 30, 2013. This work was supported by Région Languedoc-Roussillon/Feder under the Contract "Prosecure." This work is part of J. Da Rolt's Ph.D. thesis.

The authors are with the Laboratoire d'Informatique de Robotique et de Microélectronique de Montpellier, Centre National de la Recherche Scientifique, Montpellier 34392, France (e-mail: darolt@lirmm.fr; dinatale@lirmm.fr; flottes@lirmm.fr; rouzeyre@lirmm.fr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVLSI.2013.2257903

functions into the wrapper and thus considerably increases the area overhead.

Without relying on any secure protocol for accessing the test mode, some literature works have proposed to trigger a particular event when switching from mission to test mode. An essential event was first presented in [2]: a "fake" encryption/decryption key must be used during the test procedure instead of the actual secret key. With a fake key at a test time, internal states observed on the scan-out pin during the test procedure are no longer related to the secret key. This solution requires additional logic for multiplexing the secret and the "fake" test keys. In addition, the circuit's flip-flops (FFs) must be reset at the beginning of the test mode. FFs' resetting is mandatory when switching from the mission mode; otherwise, the first scan-chain unloading involved by the scan-in operation of the first test pattern reveals a circuit's state reached from the mission mode using the secret key.

FFs' resetting has been further strengthened in [10]: the reset operation is checked with the help of a multibit flag. A jump condition is thus added to each state of the circuit's FSM during scan operations, so that if the current test state has been reached from incorrect operating conditions, the reset flag indicates a wrong value and antihacking procedures can be launched. Since the reset operation cannot be checked by observing FFs' states on a scan-out pin (scan operations are not allowed before reset checking), the checking is performed using a combinational network connected to some sensitive FFs. Using the same principles, a reset operation is also performed when the circuit switches from test to mission mode for preventing data insertion via the scan path.

If we assume that an attacker can unload the scan chain without accessing the test mode (e.g., using microprobing on the scan-enable signal), he/she can thus bypass the reset operation. The solution proposed in [11] prevents such invasive attack. It consists in fixing the scan-chain structure (FFs' order) only during the test mode, while in mission mode the FFs are dynamically and randomly assigned to different position in the scan chain. This scrambling operation on the scan chain in mission mode prevent analysis of the data observed on the scan-out pin since the attacker does not know which data are observed at any moment. This solution provides a high level of security, although the mechanism for scrambling the data seriously impacts the device area and increases power consumption in mission mode.

Following the same idea, other architectures have been explored for preventing scan-based attacks by implementing "secret" function within the scan chain to obfuscate its content. The tester has to be aware of the specific hidden procedure implemented in the design, and thus, test data are first processed before being compared to expected data. In [12], inverters are inserted in the scan chain, providing bit flipping while data are scanned out. Authors in [13] have proposed the addition of XORs networks to the scan chain, providing a linear combination of test data at the scan-out instead of the test data itself. However, these solutions are all based on the assumption that the attacker has no way to get the information on the scan chain's implementation (security by obfuscation). Besides, even though these solutions have been validated for the prevention of scan attacks like the ones presented in [1] and [2], they are prone to the most recently published forms of attacks [14].

Lately, advanced DfT schemes including response compaction and X-masking techniques have been discussed to act as countermeasures [15]. The expected role of the compactor is in fact to scramble the test data in such a manner that it would be impossible to retrieve the test responses caught in the scan chain, and thus data-related secret. Unfortunately, the most recently proposed attacks [14] found a way to circumvent this type of protection.

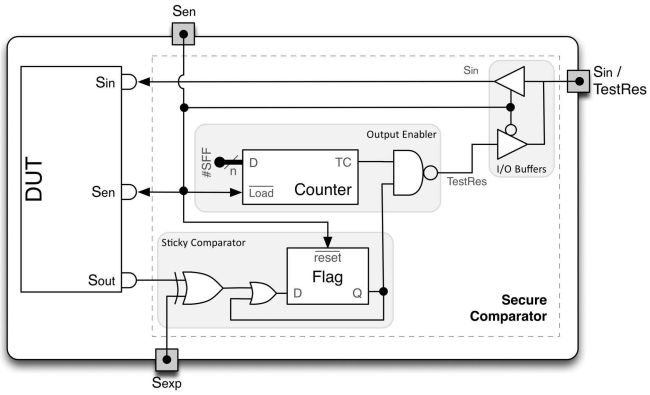

Fig. 1. Secure Comparator.

III. Proposed Solution

All scan attacks proposed in literature [1], [2], [14] rely on the possibility for the hacker to observe functional intermediate circuit states via the scan chain. Therefore, countermeasures consist in making the observation of the scan chain outputs nonexploitable.

In the standard scan-based test mechanism, FFs are replaced by scan flip-flops (SFFs) and are connected so that they behave as a shift register in test mode. The output of one SFF is connected to the input of next SFF. The input of the first FF in the chain is directly connected to an input pin (Scan-In) while the output of the last FF is directly connected to an output pin (Scan-Out). An additional signal (Scan-Enable) selects whether SFFs have to behave normally or as a shift register. The test procedure is composed of three steps: 1) test patterns are shifted-in via the scan chain (i.e., by keeping Scan-Enable $= 1$) for #SFF clock cycles (where #SFF is the number of SFFs in the chain); 2) one or two functional clocks (i.e., Scan-Enable $= 0$) are applied to capture the circuit's response. Usually, one clock cycle is used for static faults, while two (or even more) clock cycles are used for dynamic faults; 3) the content of SFFs is shifted out for #SFF clock (again, with Scan-Enable $= 1$) to allow the ATE to compare the obtained values with respect to the expected ones.

The principle of the approach proposed in this brief is to compare the actual responses with the expected ones within the chip boundaries instead of scanning-out the actual responses and comparing it within the ATE. In order to guarantee that secure data cannot leak outside the chip, the output of the comparison is not bitwise delivered to the ATE, but only after applying and comparing the whole test vector (i.e., after comparing the value of each SFF). Therefore, a potential attacker can no longer observe the FFs' content but simply pass/fail information for the whole test vector. A deeper security analysis is discussed in Section III-A.

The general scheme of the proposed Secure Comparator is shown in Fig. 1. Instead of directly shifting DUT's responses (S_{out}) out of the chip, the ATE also provides the expected responses using the S_{exp} pin and the actual test response is on-chip compared with the expected one. After having compared all #SFF bits captured in the scan chain, the signal TestRes is asserted if the whole test vector matches the one with expected values.

The Secure Comparator is composed of three parts: the Sticky Comparator responsible for the comparison between the bit stream coming from the scan chains and the expected values, and the Output Enabler triggering the final comparison result. Finally, the *I/O* Buffers allow keeping the test pin count as in a classic scan-based approach.

The Sticky Comparator performs a bitwise serial comparison between the bitstream coming from S_{out} and the one from S_{exp}.

An FF (the flag in the figure) is initially reset and then it rises to "1" whenever one comparison fails. The reset of the flag is performed when the scan operation is not enabled (i.e., S_{en} = "0"). This means then when the circuit goes from capture to test mode, the flag becomes meaningful and its value designates whether the two bitstreams are equal or not.

The Output Enabler permits the observation of the TestRes only after comparing the whole test vector. It is composed of a down counter with parallel load that loads the value #SFF whenever the scan operation is not enabled. Therefore, when the circuit goes to test mode, it starts counting and after #SFF clock cycles its terminal count allows outputting the TestRes signal through the AND gate.

The *I/O* Buffers allow sharing the same pin for S_{in} and TestRes. A classical scan-based design requires three signals: scan-in, scan-out, and scan-enable. The proposed solution requires, besides S_{in} and S_{en}, the S_{exp} signal (that replaces S_{out}) and the additional TestRes. However, S_{in} and TestRes are not used at the same time; therefore, it is possible to use bidirectional buffers shared between them, as shown in Fig. 1. During the shift operation the pin can be set as input and used by the tester to feed the circuit with the input vectors, whereas during the capture operation the pin is activated as output to deliver the previous comparison result.

Concerning the area cost, an on-chip comparison is necessary for sensitive scan chains only (the others can be treated in the usual way). However, while the Sticky Comparator is required for every scan chain, all sensitive chains can share the same counter of the Output Enabler. For a DUT with S scan chains, the longest one being composed of #SFFs, this Secure Comparator requires:

1) S flip flops and 2·S logic gates (XOR + OR) for the Sticky Comparator;
2) one counter able to count from Log_2(#SFFs) to zero, and S NAND gates to filter the TestRes signals;
3) 2·S buffers.

For example, a circuit with 32 scan chains of 10 000 SFFs each has an extra cost of 32 FFs, 98 combinational gates, 64 buffers, and one 14-bits counter. This overhead represents a negligible cost compared to the size of a circuit.

This solution does not impact the standard design flow. The secure comparator can be synthesized and connected to the S_{out} signals after circuit's synthesis and DfT insertion, without any modification to the DUT.

The same applies when the test data compression mechanisms are used to reduce the test time. In fact, since the Secure Comparator is a stand-alone module that is simply inserted after the DUT's scan-out, it can be placed downstream of the test response compactor.

IV. Security, Test, and Diagnostic Issues

This section discusses the security improvements related to the observation of a single pass/fail result as well as issues related to test and diagnosis.

A. Security Analysis

The role of the proposed Secure Comparator is to avoid the observation of SFFs containing secret information. If the result of the comparison was accessible at each clock cycle instead of each test vector, an attacker could easily observe the scan chain content by shifting in "000…000" on the S_{exp} pin. Each bit-comparison would then validate that either the actual bit was "0" when TestRes = 1 and vice versa.

On the contrary, with the proposed vector-wise comparison, the only way to retrieve the sensitive data information is to apply a brute-force attack by trying every possible response until TestRes is asserted. This attack would thus require $2^{\#SFF}$ attempts.

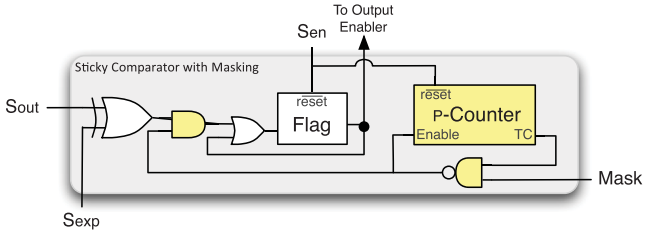

Fig. 2. Sticky Comparator with masking.

If other attacks such as side-channel attacks [16] or faults attacks [17] are dreaded, the Secure Comparator has to be protected as the rest of the circuit. Even if countermeasures can lead to a large area overhead (e.g., [18]) their implementation concerns a very small part of the circuit.

B. Testability

The secure comparator does not impact the fault coverage. In fact, each test response is compared to the expected one as in a classical ATE-based test scheme. Therefore, the achievable fault coverage is not altered. Test time is not increased either, since the expected responses are scanned-in at the same time as the next input vector is scanned-in.

Concerning the test of the Secure Comparator itself, any DfT technique controlled by the external ATE (e.g., a dedicated scan chain to test the counter of the Output Enabler) would jeopardize the overall security. Nevertheless, the Secure Comparator can be totally tested by using only its inputs (S_{en}, S_{exp}, S_{in}, TestRes). We have identified a procedure to test all stuck-at faults no matter of the size of the Secure Comparator. This functional test involves the comparison of the actual SFF values with a partially matching, a fully unmatching, and a correct response. Moreover, it includes the application of a two unmatching responses without the intermediate capture cycle, and twice the execution of the capture cycle. This test procedure requires 6· (#SFF+1) clock cycles to provide 100% stuck-at fault coverage.

A limitation of our technique is related to the presence of possible unpredictable values in the SFFs. Computing expected values for the on-chip comparison is indeed no longer possible. To fix this limitation, the Sticky Comparator should ignore the comparison result (and keep unchanged its flag) when S_{out} is unknown. This can be implemented by providing an additional mask signal that is asserted when needed. However, an attacker must not be able to mask as many bits as wanted. In fact, if it were possible to mask all but one bit, it would be obvious to discover the value of each single bit in the scan response.

This would reduce the complexity of the brute-force attack from exponential [$O(2^{\#SFF})$] to linear [O(#SFF)]. Therefore, the number of masked bit (per test vector) must be limited to P such that a brute force attack on $2^{\#SFF-P}$ remains unfeasible. The extra cost to tolerate unknown values includes an extra pin for the mask, a $\log_2$ P counter to limit the number of masked bits and two logic gates. Fig. 2 shows a possible implementation.

C. Diagnostic Ability

Limited observation of the scan chain content raises the question of whether fault diagnosis is affected. Two methods exist for identifying the cause of a faulty behavior: the effect-cause and the cause-effect. In the first case, SFF values ares analyzed together with the topology of the circuit to identify possible fault sites. This technique cannot

TABLE I
FAULT DICTIONARY AND USAGE WITH PROPOSED SCHEME

	Fault Dictionary		Proposed Method				
	v_a	v_b	v_a/r_a	v_b/r_b	v_a/r_a^{f1}	v_b/r_b^{f2}	v_b/r_b^{f3}
No Fault	r_a	r_b	1	1	0	0	0
Fault f1	r_a^{f1}	r_b	0	1	1	0	0
Fault f2	r_a	r_b^{f2}	1	0	0	1	0
Fault f3	r_a	r_b^{f3}	1	0	0	0	1
	①	②	③	④	⑤	⑥	⑦

be applied in our solution since the actual content of SFF is not delivered.

With the cause-effect approach, the circuit is fault simulated to determine the possible responses (for every input vectors) in the presence of faults. The database constructed in this step is called fault dictionary. Then the approach is based on the identification of a matching between the actual responses with those stored in the fault dictionary. If this look-up process is successful, the fault dictionary indicates the possible fault candidates. Conversely to the effect-cause approach, this process does not deal with nonmodeled faults.

Diagnostic procedure in the classical scheme, as well as with the proposed solution, is illustrated with the example in Table 1. The first row gives the data sent to the circuit, whereas the other cells indicate the result obtained from the test output pin (Scan-Out in the classic test scheme or TestRes in the proposed method). The symbol represents a test stimulus applied to the circuit through S_{in}, r_x is the corresponding expected response stored in the SFFs when no faults affect the circuit, while r_x^f is the response in the presence of the fault f. Both v_x and r_x are #SFF-bit wide. In this example, faults f2 and f3 are not exercised by the vector v_a, and fault f1 is not exercised by v_b.

In a classical test scheme, the content of the whole scan chain after application of a test vector is shifted-out to the ATE. For instance, if the collected response is r_1 when applying v_a (column ①), it means that fault f1 is not affecting the circuit. Conversely, if r_3 is observed, f1 is singly diagnosed. Similarly, when applying v_b (column ②), f2 and f3 can be differentiated according to the observed response (r_4 or r_5, respectively).

In the proposed scheme, the test of the circuit is performed by applying the pairs v_a/r_1 and v_b/r_2 by using $S_{\text{in}}/S_{\text{exp}}$ (columns ③ and ④ respectively). Since only pass/fail information is shifted-out, it is not possible to differentiate which faults cause a wrong response. For instance, a wrong response to v_a/r_1 does not allow determining whether f1 or a nonmodeled fault is present (f2 and f3 are not considered because they are not exercised by the vector v_a). In the same way, a wrong response to v_b/r_2 does not allow to distinguish among f2, f3, or even a nonmodeled fault.

In order to discriminate these cases, it is actually possible to enter, for every test vector v_x, the whole set of possible wrong-expected responses (columns ⑤, ⑥, and ⑦). For instance, if the application of v_b/r_4 provides "1" at the output of the Secure Comparator (column ⑥), the fault f2 is diagnosed.

Therefore, the proposed Secure Comparator allows the same diagnostic resolution as it can be obtained with the classical scan scheme. The only difference resides in the matching procedure between the obtained responses and those stored in the fault dictionary. In the classic scheme this is done off-line (i.e., after collecting all responses from the circuit), while in our case all faulty responses must be uploaded on the DUT, thus requiring additional time. Nevertheless, this is not an issue at diagnosis time.

V. CONCLUSION

In this brief, we proposed a novel DfT technique for scan design to ensure security without relying on costly test infrastructures to switch from mission to test modes. The proposed approach is based on the concept of withholding information. The idea is to compare test responses within the chip. Both input vectors and expected responses are scanned into the circuit and the comparison between expected and actual responses is done at vector level. It does not provide information on the value of each individual scan bit for security purposes.

Compared to regular scan test, this technique has no impact on test quality and no impact on modeled fault diagnostic. Moreover, it does not impede scan-test activities during the circuit's lifetime. The technique entails a negligible area overhead and it does not require for the designer to be particularly aware of security issues. The method can be implemented after building the scan chains, and therefore it can be applied to IP cores as well.

REFERENCES

[1] B. Yang, K. Wu, and R. Karri, "Secure scan: A design-for-test architecture for crypto chips," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 25, no. 10, pp. 2287–2293, Oct. 2006.

[2] B. Yang, K. Wu, and R. Karri, "Scan based side channel attack on dedicated hardware implementations of data encryption standard," in *Proc. IEEE Int. Test Conf.*, Oct. 2004, pp. 339–344.

[3] Y. Wu and P. MacDonald, "Testing ASICs with multiple identical cores," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 22, no. 3, pp. 327–336, Mar. 2003.

[4] K. J. Balakrishnan, G. Giles, and J. Wingfield, "Test access mechanism in the quad-core AMD opteron microprocessor," *IEEE Design Test Comput.*, vol. 26, no. 1, pp. 52–59, Jan. 2009.

[5] D. Andreu, "System and method for wirelessly testing integrated circuits," U.S. Patent 0 244 814, Oct. 6, 2011.

[6] F. Poehl, M. Beck, R. Arnold, J. Rzeha, T. Rabenalt, and M. Goessel, "On-chip evaluation, compensation and storage of scan diagnosis data," *IET Comput. Digit. Tech.*, vol. 1, no. 3, pp. 207–212, 2007.

[7] G.-M. Chiu and J. C.-M. Li, "A secure test wrapper design against internal and boundary scan attacks for embedded cores," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 20, no. 1, pp. 126–134, Jan. 2012.

[8] K. Rosenfeld and R. Karri, "Attacks and defenses for JTAG," *IEEE Design Test Comput.*, vol. 27, no. 1, pp. 36–47, Jan. 2010.

[9] C. J. Clark, "Anti-tamper JTAG TAP design enables DRM to JTAG registers and P1687 on-chip instruments," in *Proc. IEEE Int. Symp. Hardw.-Oriented Security Trust*, Jun. 2010, pp. 19–24.

[10] D. Hely, F. Bancel, N. Berard, M. L. Flottes, and B. Rouzeyre, "Test control for secure scan designs," in *Proc. IEEE Eur. Test Symp.*, May 2005, pp. 190–195.

[11] D. Hely, M.-L. Flottes, F. Bancel, B. Rouzeyre, N. Berard, and M. Renovell, "Scan design and secure chip [secure IC testing]," in *Proc. IEEE Int. On-Line Test. Symp.*, Jul. 2004, pp. 219–224.

[12] G. Sengar, D. Mukhopadhyay, and D. R. Chowdhury, "Secured flipped scan-chain model for crypto-architecture," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 26, no. 11, pp. 2080–2084, Nov. 2007.

[13] H. Fujiwara and M. E. J. Obien, "Secure and testable scan design using extended de Bruijn graphs," in *Proc. Asia South Pacific Design Autom. Conf.*, 2010, pp. 413–418.

[14] J. Da Rolt, G. Di Natale, M.-L. Flottes, and B. Rouzeyre, "New security threats against chips containing scan chain structures," in *Proc. IEEE Int. Symp. Hardw.-Oriented Security Trust*, Jun. 2011, pp. 110–115.

[15] L. Chunsheng and Y. Huang, "Effects of embedded decompression and compaction architectures on side-channel attack resistance," in *Proc. IEEE VLSI Test Symp.*, May 2007, pp. 461–468.

[16] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Proc. Int. Cryptol. Conf. Adv. Cryptol.*, 1999, pp. 388–397.

[17] P. Dusart, G. Letourneux, and O. Vivolo, "Differential fault analysis on A.E.S," in *Applied Cryptography and Network Security*, vol. 2846. New York, NY, USA: Springer-Verlag, 2003, pp. 293–306.

[18] A. Moradi, T. Eisenbarth, A. Poschmann, C. Rolfes, C. Paar, M. T. M. Shalmani, and M. Salmasizadeh, "Information leakage of flip-flops in DPA-resistant logic styles," in *Proc. IACR Cryptology ePrint Archive*, 2008, pp. 188–188.