Partial Core Encryption for Performance-Efficient Test of SOCs *

Ozgur Sinanoglu and Alex Orailoglu

Computer Science and Engineering Department
University of California, San Diego
La Jolla, CA 92093

{ozgur, alex}@cs.ucsd.edu

ABSTRACT

The isolation of a core through full I/O scan helps ease SOC test challenges; yet the performance of high-speed SOCs is significantly hampered. We propose a partial core encryption methodology wherein the core vendor unveils only a small part of the core logic, successfully satisfying core IP protection requirements. Once the partially encrypted cores are merged into an SOC, the system integrator performs test generation on the visible SOC logic only, greatly reducing the test generation effort expended. By utilizing the test data provided by the core vendor as well, the SOC integrator can test the SOC with no performance degradation. We present an efficient fault analysis based core encryption algorithm which is guided by judiciously computed testability measures. The experimental results confirm the significantly high encryption levels attained by the proposed encryption algorithm.

1. INTRODUCTION

Stringent design performance requirements magnify the impact to performance degradation induced by Design For Test (DFT) techniques. Logic gate insertion on critical paths for enhancing the testability of the circuits impacts their functional operation timing-wise, undermining the expected fulfillment of the stringent performance requirements.

System-On-a-Chip (SOC) designs suffer from performance degradation particularly in the case of unmergeable cores. While design and test are simplified due to the reuse strategy employed for both, full encryption of core logic necessitates the isolation of cores during test. The consequent lack of structural information regarding the core logic forces the full scan of all core I/Os by the system integrator so as to ensure the delivery and collection of test data to and from the core I/Os during test application. In this scheme, the core vendor provides not only the fully encrypted core but also the test data for the core. The test of the core is significantly simplified, consequently, regardless of how deeply the core is embedded in the SOC. Despite the beneficial test enhancements, the additional gate delays inserted on the core boundary result in a timing impact on the functional operation of the SOC.

Mergeable cores, on the other hand, result in no performance degradation. As the core logic is not encrypted and hence is fully exposed, the SOC integrator performs test generation by treating the SOC as a monolithic entity. In this scheme, no need for scanning the core I/Os exists, as the necessity for core isolation during test is eliminated; no interference with the SOC functional operation timing-wise ensues. The downside, however, is that the test challenges are offloaded onto the system integrator, as test generation needs to be performed from scratch for SOCs of considerable size.

In order to eliminate the performance drawbacks of full core logic encryption and the test challenges associated with full core logic exposure, we propose a partial core encryption methodology in this paper. In the proposed scheme, a significant part of the core logic is concealed from the core user, while the test of the concealed logic is ensured through the test data provided to the SOC integrator. The SOC integrator utilizes this test data for testing the majority of the faults in the core; both the remaining faults and the logic that should be unveiled to the core user for testing these faults are provided to the SOC integrator. While the core IP is significantly preserved, the necessity for core isolation during test is eliminated in the proposed scheme, resulting in no performance degradation. Furthermore, only a small fraction of the SOC is targeted in test generation, significantly reducing the test generation effort expended by the system integrator.

Although the elimination of core isolation during test delivers performance benefits, it reflects into hampered core I/O access. The effectiveness of the test data provided to the SOC integrator is limited, consequently, resulting in typically a fraction of irredundant core faults remaining untestable. To generate test stimuli for these faults, the SOC integrator needs to access the required parts of the core logic. To account for the stringent IP protection concerns particularly of the vendors that specialize in unmergeable cores, we present an efficient *fault analysis based* encryption technique that delivers very high encryption levels. We also define testability measures to judiciously guide the proposed encryption technique, minimizing the amount of logic unveiled to the SOC integrator.

2. PREVIOUS WORK

In recent years, a considerable amount of effort has been expended for testing core-based SOCs. Embedded core test challenges and a survey of the previously published approaches can be found in [1].

Numerous techniques have focused on reducing performance degradation due to core I/O scan. The partial isolation ring technique in [2] proposes the scan of a subset of core I/Os. The test responses of the other cores propagated to the inputs of the core under test are utilized for obtaining the desired core test stimuli, eliminating the scan of a number of core inputs. Similarly, core test responses are utilized to obtain the stimuli for testing the neighboring core(s), enabling partial scan of the core outputs. Performance degradation of this technique can be severe depending on the number of scanned core I/Os. In [3], the utilization of a new type of storage element is proposed for the core outputs; both the observation of the core responses and the justification of the stimuli for testing the neighboring core(s) can be performed at significantly reduced performance cost. However, the technique still suffers from performance degradation due to the multiplexer insertion at the core inputs. A transparency-based test generation technique is proposed for core-based SOCs in [4]. For every core, justification and sensitization paths through other cores are constructed by inserting test points, enabling the translation of core-level stimulus to system-level stimulus. Performance degradation can still occur in this technique due to the insertion of numerous test points. Furthermore, challenges in transparency channel construction limit the applicability of this technique to cores

*The work of the first author is supported through an IBM graduate fellowship.

ICCAD'03, November 11-13, 2003, San Jose, California, USA.

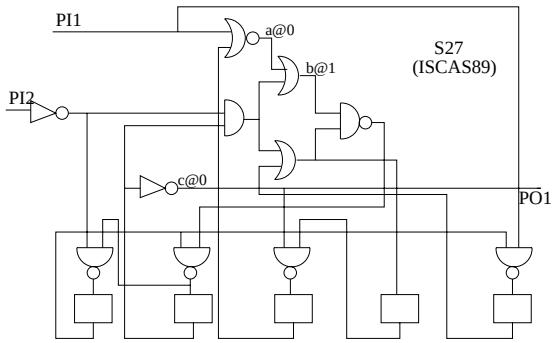

Figure 1: Scan-testable and scan-untestable faults

such as digital signal processors and microprocessors.

Encryption of soft cores has been proposed in [5]; the logic that can be tested solely through scan inputs is concealed from the core user. The approach proposed in [5] constitutes a good starting point but is rather pessimistic. Not only the logic cone driving the primary outputs but furthermore the logic cone driving the pseudo-primary outputs that are reachable by any primary input are completely unveiled, resulting in the concealment of only a small set of gates in the core. The algorithmic weakness of this scheme hinges on the exposition of the complete logic cone with no judicious justification analysis. This pessimistic approach requires, for instance, the exposure of the complete logic driving a gate input through which a fault is to be propagated, whereas the unveiling of a much smaller number of gates in the logic cone might well have sufficed for the justification of the gate input to the non-controlling value. Insufficient encryption levels delivered by this scheme, typically 20-25% of the gates on the average as reported in the paper, hamper its applicability to hard cores. Satisfactory encryption levels can be somewhat ensured by this technique only if the core I/Os are partially scanned, in a manner similar to the technique in [2], albeit at the cost of appreciable performance degradation.

3. SCAN TESTABILITY

In this section we identify certain test information about a core that is preserved, regardless of how deeply the core is embedded in an SOC. We propose a scheme wherein the core vendor provides this information along with the partially encrypted core to the SOC integrator. The test of the SOC embedding the core is thus enabled with no fault coverage loss and no performance degradation.

While full access to both primary inputs and primary outputs can be assumed in a standalone test of a core, the reduced access in a core embedded in an SOC complicates the core test. Faults untestable in a core being tested standalone with full access of course remain untestable once the core is embedded. Providing the set of these untestable core faults to the SOC integrator could thus help save time and effort by avoiding test generation for untestable faults.

Regardless of how limited core I/O access is, the core scan cells can be fully accessed during the test of the SOC that embeds the core. The faults that can be detected during the test of a standalone core solely through the scan path with no control of primary inputs and no observation of primary outputs can still be detected during the test of the SOC. We denote the faults that can be tested solely through scan cells as *scan-testable faults*, and the test stimuli to be delivered into the scan cells for testing the scan-testable faults as *scan-stimuli*. For instance in Figure 1, the stuck-at-1 fault, $b@1$, can be detected with no control of primary inputs and no observation of primary outputs; the scan-stimulus that sensitizes this fault to the second scan cell from the left is "101x1". The stuck-at-0 fault, $a@0$, in Figure 1 is scan-untestable as it can be detected only if $PI1$ can be controlled to 0. It should be noted that the proposed scheme is independent of the particular fault model; alternate fault types, such as bridging faults for instance, can be included in the scan-testability analysis as well. In the proposed scheme, core scan-stimuli are provided by the core vendor; during the test generation for the SOC, the SOC integrator targets therefore the scan-untestable faults, except for the originally untestable ones, easing the test generation process significantly.

The faults that are scan-untestable due to the lack of core primary output observation can be remedied in a cost-effective manner. The connection of a compaction circuitry, such as a parity tree or a MISR, to the core outputs restores the primary output observation capabilities, enhancing the scan-testability of the core. As the compaction circuitry is inserted on the test path, it has no impact on functional operation timing-wise, resulting in no performance degradation. The only cost of such a scan-testability enhancement strategy consists of the area overhead.

We propose a scheme wherein the SOC integrator needs to target only the SOC scan-untestable faults during test generation for the SOC. The SOC integrator needs to access only the logic that suffices to generate test patterns for the scan-untestable faults, enabling a scheme wherein a partially encrypted core is provided by the core vendor. The responsibility of the core vendor also includes the delivery of the scan-stimuli which are to be inserted into the scan cells of the cores during test application. Partially encrypted cores enable core vendors to preserve their IPs, while they help SOC integrators eliminate performance degrading I/O scan and ease test generation. A partially encrypted core scheme combines the advantages of both unmergeable and mergeable cores, consequently. Having all the information regarding the critical paths of the system, the SOC integrator is the one who can handle the SOC testability problems without incurring any performance degradation. The proposed scheme is quite beneficial as it enables the SOC integrator to fix testability problems, if any, at the system-level without being forced to employ *a priori* any performance degrading techniques such as full I/O scan. The testability of the User Defined Logic (UDL) surrounding IP cores, for instance, can be ensured by the system integrator through the judicious insertion of testability points with no critical path violation.

4. ALGORITHMIC FRAMEWORK

To ensure the applicability of the proposed scheme, particularly to hard cores, the core encryption level attained should be satisfactory to the core vendor. The core encryption algorithm should therefore be able to conceal the maximal number of gates based on a set of faults for which test generation is to be performed. In this section we present the proposed fault analysis based core encryption algorithm to be employed by core vendors. We first introduce a symbolic controllability analysis to efficiently identify the observation points that significantly enhance the core encryption level with no performance degradation. Subsequently, we define encryption measures for guiding the proposed algorithm.

4.1 Symbolic Controllability Measures for Observation Point Insertion

To identify, in a computationally efficient manner, the parts of the core that cannot be tested through the control of scan cells, we introduce a symbolic controllability analysis. Such an analysis enables the core vendor to enhance the scan-testability of the core, resulting in higher levels of core encryption. Observability points can be inserted inside the core based on the proposed symbolic controllability analysis, enabling the concealment of the logic further. The observability points are connected to the compaction circuitry along with the primary outputs, attaining significant encryption level enhancements at no performance cost.

The symbols we utilize consist of c_ϕ, c_0, c_1, c_u denoting the circuit lines that are controllable neither to 0 nor to 1, controllable to 0 only, controllable to 1 only and controllable to both 0 and 1 values, respectively. In the computation of these symbolic measures for the circuit lines, initially the primary inputs of the circuit are all set to c_ϕ whereas

	c_ϕ	c_{cont}	$c_{\overline{cont}}$	c_u
c_ϕ	c_ϕ	$c_{cont \oplus inv}$	c_ϕ	$c_{cont \oplus inv}$
c_{cont}	$c_{cont \oplus inv}$	$c_{cont \oplus inv}$	$c_{cont \oplus inv}$	$c_{cont \oplus inv}$
$c_{\overline{cont}}$	c_ϕ	$c_{cont \oplus inv}$	$c_{\overline{cont \oplus inv}}$	c_u
c_u	$c_{cont \oplus inv}$	$c_{cont \oplus inv}$	c_u	c_u

cont: controlling value, inv: inversion parity

Table 1: Controllability symbol propagation rules

the lines controlled by the scan elements are initialized to c_u. The propagation rules for these symbols can be defined in a generalized form for all logic gates as in Table 1 wherein the propagation through an arbitrary gate is modeled based on the gate controlling value, denoted as $cont$, and the gate inversion parity, denoted as inv. Given that the symbols corresponding to a row and a column appear at the inputs of a gate, the corresponding entry constitutes the symbol propagated to the output of the gate. Scan-testability is strongly correlated to the number of c_u symbols in the circuit, as these symbols originate from the scan elements and denote the logic which is fully controllable by the scan cells.

Figure 2 illustrates an example circuit with the computed symbolic controllability values. It can be seen that the outputs of the gates which are controlled by only the primary inputs ($g1$) are set to c_ϕ while the outputs of the ones controlled only by the scan cells ($g5$, $g6$, $g7$) are c_u. The controllability level of the gates that are controlled by both primary inputs and scan cells, on the other hand, depends on the other gates (controlling value and inversion parity) in the logic cone. For instance, the output of $g2$ is controllable to 0 as one of its inputs is controllable to 0. Although $g4$ is controlled by primary inputs as well, both of its inputs are fully controllable.

The gates with at least one not fully controllable input, such as $g1$, $g2$, and $g3$, need to be unveiled as they contain scan-untestable faults. The gates ($g4$, $g8$) that happen to be on the propagation path for such faults can still be concealed with the insertion of an observation point (at the output of $g3$); these observation points help contain the encryption degrading impact of the primary inputs. The technique we propose can be contrasted with previous techniques, such as [5], which fail to hide any of the gates for the example given in Figure 2.

4.2 Encryption Measures

In order to guide the encryption algorithm, we define encryption measures in terms of controllability and observability costs. The overall goal of the encryption algorithm is the minimization of the logic gates to be unveiled. The test of multiple scan-untestable faults using the same set of gates apparently serves this goal; in case multiple justification (observation) paths exist for testing a scan-untestable fault, the path that has already been unveiled for testing other faults should be selected. The encryption measures we define herein enable the encryption algorithm to select appropriate paths.

We define the testability cost of a line as the number of gates to be unveiled; for instance, the cost of controlling a line to 0 denotes the number of gates to be unveiled so as to justify this line to 0. Once the primary input/scan cell controllability costs and the primary output/scan cell ob-

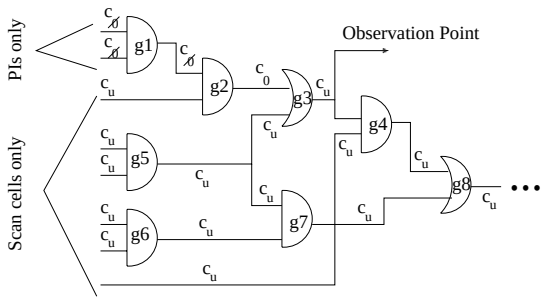

Figure 2: Symbolic controllability values

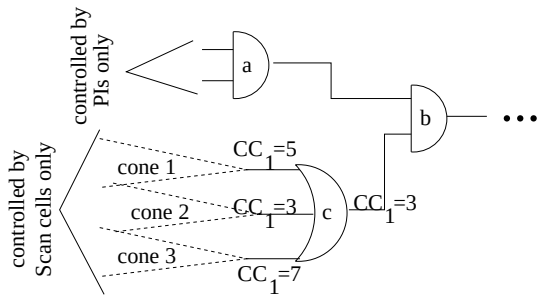

Figure 3: Encryption measures in terms of testability costs

servability costs are all set to 0, the controllability and observability costs of internal lines can be computed based on the following formula:

$$\begin{aligned} CC_{cont \oplus inv}(out) &= min(CC_{cont}(in1), CC_{cont}(in2)) \\ CC_{\overline{cont \oplus inv}}(out) &= CC_{\overline{cont}}(in1) + CC_{\overline{cont}}(in2) \\ OC(in1) &= OC(out) + CC_{\overline{cont}}(in2), \end{aligned}$$

wherein $cont$ and inv denote the controlling value and the inversion parity of the gate, and $CC_v(signal)$ and $OC(signal)$, the controllability to v and the observability cost of $signal$, respectively. The formula derived for a gate with two inputs ($in1$ and $in2$) and one output (out) can be easily generalized for gates with more than two inputs. One can note from the formula that the controllability costs should be computed prior to the observability costs. Furthermore, forward logic traversal is performed in computing the controllability costs whereas the computation of observability costs necessitates backward logic traversal.

The example circuit in Figure 3 illustrates the benefit of the encryption measures utilized. Testing the faults in gate a, which is controlled by primary inputs, necessitates the unveiling of gate b, enabling the propagation of the effects of these faults. Furthermore, to avoid blocking these fault effects, the output of gate c should be justified to 1. Two of the three logic cones can still be concealed as only one input of gate c should be justified to 1. The controllability cost of the gate c inputs can be compared to select the input to be justified to 1, enabling the selection of one logic cone with the minimal number of gates to be unveiled, namely cone 2 in Figure 3. It should be noted that the selected input of gate c can still be justified to 1 by unveiling *partially* the logic cone driving it.

4.3 Core Encryption Algorithm

Subsequent to the insertion of the observation points identified through the utilization of the symbolic controllability values, the testability cost directed encryption algorithm is executed based on the set of faults that are scan-untestable. The algorithm is dynamic as throughout the course of its execution, the testability costs are continuously updated based on the gates that have already been unveiled. These encryption measures guide the selection of not only the justification (sensitization) paths but furthermore the target fault to be handled as well.

The scan-untestable faults in a core can be easily identified through the execution of test generation with the core primary inputs tied to *don't care* values. If a compaction circuitry is utilized, it should be connected to the primary outputs during test generation; otherwise, the observation of the primary outputs is disabled. The faults that remain undetected upon termination of the test generation process constitute the scan-untestable faults. The faults that the core encryption algorithm is executed on consist of the scan-untestable faults which are originally detectable.

In the encryption algorithm, the gates with at least one scan-untestable fault are unveiled initially. Furthermore, the scan-untestable faults with a unique sensitization or justification path necessitate the unveiling of the gates on these paths as well. Subsequently, the testability costs are com-

Circuit	PI	PO	SC	Encryption Level (%)			
				OP=0	OP=3	OP=6	OP=10
s13207	31	121	669	90.1	91.6	92.3	92.5
s15850	14	87	597	55.6	62.7	66.8	70.3
s35932	35	320	1728	96.2	96.8	97.1	97.2
s38417	28	106	1636	82.6	84.5	85.3	86.1
s38584	12	278	1452	83.0	85.2	86.7	87.8
Avg.	24	182	1216	81.5	84.2	85.7	86.8

Table 2: Encryption levels attained by the proposed methodology

puted by taking into account the gates that have already been unveiled; these gates have no contribution to the testability costs as the algorithm tries to minimize greedily the additional number of gates to be unveiled. For each scan-untestable fault, an estimation of the number of gates to be unveiled for testing is obtained by adding up the controllability and the observability costs of the fault; the fault with the minimal cost is targeted. In case multiple justification and/or sensitization paths exist for this fault, the testability costs are utilized for selecting the path(s) with the minimal cost. Subsequent to handling the fault, the testability costs are recomputed, reflecting the impact of the gates that have already been unveiled. The algorithm terminates when all the scan-untestable faults are handled. Upon the termination of the algorithm, test generation can be performed on the partially encrypted core as a verification step so as to ensure the testability of all the scan-untestable faults.

5. EXPERIMENTAL RESULTS

The proposed core encryption methodology has been applied to the five largest fully-scanned circuits in ISCAS89 [6]. The ATPG tool, ATALANTA [7], has been utilized for generating scan-stimuli and identifying scan-untestable faults. In this section, the encryption levels attained by the proposed methodology along with the comparisons against the core encryption technique in [5] are provided.

Table 2 demonstrates the encryption levels achieved by the proposed scheme. The number of circuit inputs, outputs, and scan cells are provided in columns 2, 3, and 4, respectively, whereas columns 5 through 8 report the encryption levels attained by the proposed algorithm when 0, 3, 6, and 10 observation points are inserted, respectively. The encryption levels are calculated as the percentage of gates that can be hidden in the circuit. It can be noted from the results that as more observation points are inserted, an increased number of gates can be hidden; however, the number of observation points is typically limited as additional lines to be observed typically complicate the compaction circuitry. Even with no observation points inserted, significantly high encryption levels are attained by the encryption methodology we propose. For four of the five benchmark circuits, an encryption level in excess of 80% is attained; an encryption level of 96.2% is obtained for s35932. For s15850, however, 55% of the gates only can be hidden; with the insertion of a few observation points, the encryption level for this circuit can be brought to a satisfactory level with no performance degradation whatsoever.

Encryption level comparisons against the logic cone based technique[1] in [5] are provided in Table 3. As the encryption level results reported in [5] are computed with no testability point insertion, we provide the encryption levels attained by the proposed methodology with the observation point insertion capabilities turned off, thus comparing the two schemes on an even basis. As expected, the scheme we propose outperforms the technique in [5] for all five of the circuits. Encryption levels in excess of 30% are achieved for only two circuits in [5], sharply limiting its practical applicability. Dramatic encryption level gaps between the proposed methodology and the one in [5] exist for s35932 and s38584;

[1] We have re-implemented the technique in [5] since the experiments reported therein have been performed on internally available and undisclosed industrial benchmark circuits.

Circuit	Logic Cone Analysis (%) [5]	Proposed (%)
s13207	62.7	90.1
s15850	29.6	55.6
s35932	16.6	96.2
s38417	66.4	82.6
s38584	2.1	83.0
Avg.	35.5	81.5

Table 3: Encryption level comparisons

while 16.6% and 2.1% encryption levels are attained by [5], the methodology we propose achieves 96.2% and 83.0% encryption. The results in Table 3 justify the benefit of the utilization of a fault-based methodology rather than the previously suggested and pessimistic logic cone based approach for encrypting a circuit.

6. CONCLUSION

In this paper, we propose a partially encrypted core scheme wherein the core logic is made partially visible to the SOC integrator, preserving the core IP. The responsibility of the core vendor includes delivery not only of the partially encrypted core but furthermore of the scan-stimuli for testing the scan-testable faults. Subsequent to merging the partially encrypted cores into an SOC, the SOC integrator performs test generation on the visible logic for only the scan-untestable faults. The test set to be applied to the SOC consists of both the stimuli generated for scan-untestable faults and the system level scan-stimuli formed by concatenating the core-level stimuli. As the necessity for I/O scan is eliminated, SOCs with partially encrypted cores can be tested with no performance degradation. Furthermore, as only scan-untestable faults are targeted, test generation time expended by the SOC integrator is significantly reduced.

We also define a symbolic controllability analysis which helps identify the observation points that increase the core encryption level. Furthermore, we propose a fault analysis based core encryption algorithm, which is directed by the encryption measures defined. The utilization of these encryption measures enables the unveiling of the same set of gates for testing various scan-untestable faults, yielding a significantly high core encryption level.

The experimental results confirm the significantly high encryption levels attained by the proposed algorithm. The set of methodologies we present in this paper enable high quality, IP-preserving SOC tests while the increasing performance requirements of SOCs are fulfilled at the same time.

Acknowledgement

The authors would like to thank Brion Keller and Vivek Chickermane from Cadence for their insightful comments and suggestions.

7. REFERENCES

[1] Y. Zorian, E. J. Marinissen and S. Dey, "Testing Embedded-Core Based System Chips", in *ITC*, pp. 130–143, 1998.

[2] N. A. Touba and B. Pouya, "Using Partial Isolation Rings to Test Core-Based Designs", *IEEE Design and Test of Computers*, pp. 52–59, October 1997.

[3] S. Bhatia, T. Gheewala and P. Varma, "A Unifying Methodology for Intellectual Property and Custom Logic Testing", in *ITC*, pp. 639–648, 1996.

[4] I. Ghosh, N. K. Jha and S. Dey, "A Low Overhead Design for Testability and Test Generation Technique for Core-Based Systems-on-a-Chip", *IEEE TCAD*, vol. 18, n. 11, pp. 1661–1676, November 1999.

[5] K. De, "Test Methodology for Embedded Cores which Protects Intellectual Property", in *VTS*, pp. 2–9, 1997.

[6] F. Brglez, D. Bryan and K. Kozminski, "Combinational Profiles of Sequential Benchmark Circuits", *IEEE ISCAS*, vol. 14, n. 2, pp. 1929–1934, May 1989.

[7] H. K. Lee and D. S. Ha, *On the Generation of Test Patterns for Combinational Circuits*, Technical Report 12-93, Department of Electrical Eng., Virginia Polytechnic Institute and State University.