Efficient and Secure Split Manufacturing via Obfuscated Built-In Self-Authentication

Kan Xiao, Domenic Forte and Mark (Mohammed) Tehranipoor
Department of Electrical & Computer Engineering
University of Connecticut
{kanxiao},{forte},{tehrani}@engr.uconn.edu

Abstract

The threats of reverse-engineering, IP piracy, and hardware Trojan insertion in the semiconductor supply chain are greater today than ever before. Split manufacturing has emerged as a viable approach to protect integrated circuits (ICs) fabricated in untrusted foundries, but has high cost and/or high performance overhead. Furthermore, split manufacturing cannot fully prevent untargeted hardware Trojan insertions. In this paper, we propose to insert additional functional circuitry called obfuscated built-in self-authentication (OBISA) in the chip layout with split manufacturing process, in order to prevent reverse-engineering and further prevent hardware Trojan insertion. Self-tests are performed to authenticate the trustworthiness of the OBISA circuitry. The OBISA circuit is connected to original design in order to increase the strength of obfuscation, thereby allowing a higher layer split and lower overall cost. Additional fan-outs are created in OBISA circuitry to improve obfuscation without losing testability. Our proposed gating mechanism and net selection method can ensure negligible overhead in terms of area, timing, and dynamic power. Experimental results demonstrate the effectiveness of the proposed technique in several benchmark circuits.

I. Introduction

The integrated circuits (ICs) supply chain has changed significantly over the last decade as semiconductor scaling has reached very deep submicron levels. The new advanced and complex semiconductor technology requires prohibitive investment. Thus, most semiconductor companies now outsource fabrication to off-shore state-of-the-art foundries. Although semiconductor companies can lower manufacturing cost and speed up the development cycle by outsourcing fabrication or design, they gradually lose the control of the whole supply chain. Untrusted third-party foundries could reverse-engineer the layout design, pirate IP and even add malicious circuitries, i.e., hardware Trojans. This is a difficult situation for the IC design companies who not only wish to push performance to the edge by using off-shore state-of-the-art technologies but also want to guarantee security for critical applications.

Split manufacturing, or split fabrication (used interchangeably in this paper), has been proposed recently as an approach to enable use of state-of-the-art semiconductor foundries while minimizing the risks to IP [1] [2]. Split manufacturing divides a design into Front End of Line (FEOL) and Back End of Line (BEOL) portions for fabrication by different foundries. An untrusted foundry performs (higher cost) FEOL manufacturing, then ships wafers to a trusted foundry for (lower cost) BEOL fabrication.

Two types of split manufacturing have been proposed in prior work: 2D integration and 3D integration based split fabrication. [3] first proposed the use of 3D integration of two tiers (the computation plane and the control plane) manufactured in separate foundries to ensure the performance of the computation plane and the security of the control plane. One tier is stacked on the top of another tier and conventional 3D stacked integration techniques are required to merge two tiers with vertical interconnections called through-silicon-vias (TSVs). Unfortunately, the semiconductor industry has not adopted 3D ICs as quickly as many in the industry expected. Given the barriers to 3D, 2D and 2.5D based split manufacturing are discussed more, such as those in [4] [5] [6] [7] [9]. [10] developed a technique that makes FEOL and BEOL fabricated separately in different foundries and can make connections between them with wafer-bonding at a fine enough pitch similar to TSVs. [4] and [5] demonstrated the feasibility of split fabrication after metal 1 (M1) on test chips and evaluated the chip performance. Although the split after M1 attempts to hide all inter-cell interconnections and can obfuscate the design effectively, it leads to high manufacturing costs. [6] and [7] employed another integration approach. The back-end layers can be manufactured directly on top of the front-end layers with mask alignment techniques in a trusted foundry, which was studied on an FPGA chip fabricated in a split manufacturing process [7]. Finally, [8] presents a *k*-security metric to select wires to be lifted to a trusted tier (BEOL) to ensure the security when split at a higher layer. However, lifting a large number of wires in the original design will introduce large timing($\geq$73%) and power($\geq$54%) overhead and significantly impact chip performance [8], since delay and power are strong functions of wire length. In addition, though *k* similar elements can be created by lifting sufficient wires, it cannot prevent adversaries from tampering all these similar elements with untargeted Trojans.

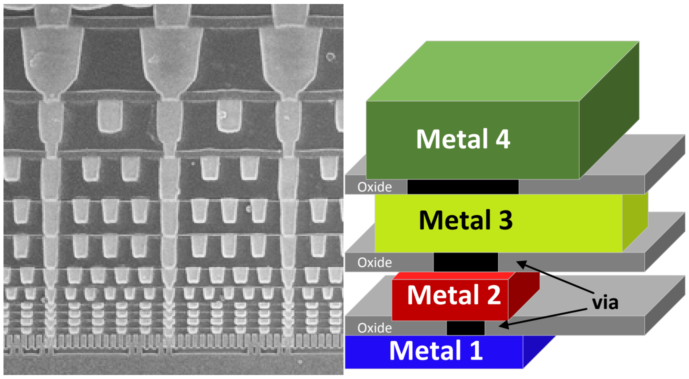
In this paper, we propose a new design methodology that can effectively prevent reverse engineering of the chip functionality and further prevent hardware Trojan insertion with split fabrication process. Our technique allows FEOL and BEOL to be separated at higher layers ($\geq$ M4) to reduce cost. In order to enhance effectiveness of obfuscation, all unused spaces in layout will be filled with additional functional cells or circuitry called obfuscated built-in self-authentication (OBISA) instead of non-functional filler cells during layout design. If any of the OBISA cells are replaced by a Trojan cell during FEOL, OBISA will be able to detect the modification. We propose to make connections between OBISA added into unused spaces and the original circuit, especially in its critical parts that are to be protected. The OBISA circuit not only makes it extremely difficult for adversary to identify the original design, but also thwarts hardware Trojan insertion by filling unused spaces in layout. In addition, several design-for-security methods are proposed to minimize timing and power overhead introduced by OBISA circuit while maintaining a high test coverage for OBISA.

The rest of the paper is organized as follows: Section II presents the tradeoff between low-layer and high-layer splits. In Section III, we will present OBISA technique with split fabrication. In Section IV, we will propose our placement and routing strategies for OBISA circuits. Implementations and results will be presented in Section V. Finally, Section VI concludes the paper.

II. Low-Layer Split vs. High-Layer Split

A. Cost Issues

Reverse engineering requires analysis of local standard-cell types and their interconnections in order to identify structures or some targeted logics within a circuit. Split fabrication can prevent reverse-engineering the complete design or macro in a layout design by hiding a portion of wires. However, an adversary still could identify some sub-circuits, such as adder, decoder, cryptographic logic, etc., based on FEOL mask-layer information. The strength of obfuscation depends on the split layer, i.e., the layer that ends at the FEOL, since it determines how much layout information will be exposed to untrusted foundries. Figure 1 (a) shows a cross-section of a 14*nm* Intel chip [17]. A high-layer split leaks more interconnections while low-layer split leaks much less. Split after M1 provides exceptional circuit obfuscation, because adversaries at untrusted foundry only see unconnected gates in layout and no inter-cell connections at all [4] [5]. However, such a low-layer split also brings challenges in split manufacturing process.



(a) Intel's 14nm Chip [17]

(b) Metal layers and vias

Figure 1: A cross-section of an IC.

First, high metal layers are thicker and have a larger pitch than low metal layers, as shown in Figure 1 (a) and (b). Table I shows the pitches of different metal layers for a 45nm technology [9] [16]. The integration process of FEOL and BEOL wafers requires precise alignment using either electrical, mechanical, or optical alignment techniques. A via requires a certain amount of space surrounding in order to satisfy design rules. If alignment is not perfect, the misalignment defects could influence circuit performance or even produce malfunction. However, the tolerance for misalignment improves greatly if the split occurs at higher layers. Unfortunately, misalignment has a higher probability to occur in split fabrication due to different process technologies or facilities at two different foundries (FEOL and BEOL), the yield for low-layer split could be relatively low.

Table I: Pitch length of different metal layers in 45nm CMOS technology [9].

Layer	Poly	M1	M2	M3	M4	M5	M6	M7	M8	M9
Pitch(nm)	125	130	140	140	280	280	280	800	800	1600

Second, higher layer split leads to fewer and less dense connections between the trusted and untrusted tiers, and can mitigate the challenges of alignment for a large number of connections. Therefore, high-layer splits would reduce complexity of integrating FEOL and BEOL and result in a high integration yield.

Third, lower layer splits also requires a closer technology match between foundries. One major reason for choosing split fabrication is that trusted foundry (BEOL) has less advanced technology and cannot meet the specification for a particular design. A split at a higher level can allow BEOL fabricated with older process technologies, making split manufacturing more widely acceptable and further reducing its costs.

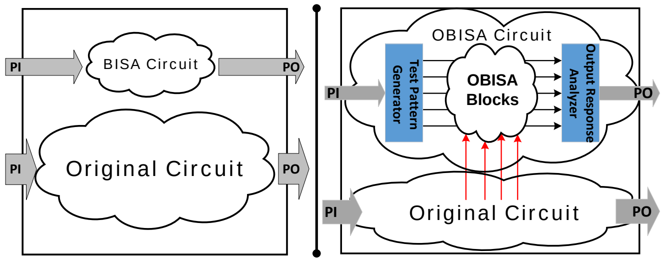
B. Security Issues: Obfuscation and Tampering

Split at a higher layer has many pros as described above, but it also results in significant interconnections for circuit blocks information leakage. Although adversaries cannot reverse engineer the entire design due to lack of connections in BEOL, it still provides adversaries opportunities to identify some sub-circuits (such as adder, decoder, and FSM) and tamper them with hardware Trojans [13]. [4] and [6] proposed to insert additional "dummy" cells as spare cells for obfuscating the composition of a circuit, but the inserted cells have no interconnections between themselves or between them and the main design. Hence, they are easy to identify.

III. Split Fabrication with OBISA

A. Proposed Approach

Based on the previous techniques, there is a tradeoff between cost and security for split manufacturing. However, our objective is to develop a methodology that allows high-layer splits, M3 or higher, while lowering the cost and at the same time providing a high level of security. We propose to insert *functional standard cells* into unused spaces of layout instead of filler cells or dummy cells. The inserted cells are connected together to form a circuitry, called obfuscated built-in self-authentication (OBISA). Note that this is different from the previously proposed built-in self-authentication (BISA) [11] [12] for preventing Trojan insertion both from design and objective standpoints. As shown in Figure 2 and 3, OBISA is connected to the original circuit it is trying to protect, which makes it extremely difficult for adversaries at untrusted foundry to separate the OBISA design from the original design. However, OBISA circuit would result in dynamic power and timing overhead when the original circuit is operating. Therefore, a gating mechanism and a net selection method are proposed to minimize the negative impact on the original design. Additionally, we attempt to maintain a high test coverage for OBISA circuit with tree-structure circuits as in [11] [12]. Its built-in self-test (BIST) like structure can detect potential malicious modifications by test patterns that target stuck-at faults. However, the tree-structure circuit and test structure components could become a target for the adversary to identify them. In this paper, an approach is presented that can create fan-outs to further obfuscate tree-structure circuit in OBISA without impacting its original high test coverage. Moreover, lifting critical wires to the trusted tier (BEOL) can prevent identifying the test structure within OBISA.



(a)

(b)

Figure 2: (a) BISA structure in [11] and (b) the proposed OBISA structure for split manufacturing in this paper.

B. OBISA structure

Figure 2(b) shows the structure of the proposed OBISA. It consists of a test pattern generator (TPG), an output response analyzer (ORA) and OBISA blocks under test. The entire OBISA circuit is implemented in unused spaces of the layout. In this paper, we use a linear feedback shift register (LFSR) as TPG and multiple input shift register (MISR) as ORA. They are used to generate random test patterns and compress responses to generate a final signature, respectively [15].

Two operating modes are associated with the proposed technique. In *functional mode*, the original circuit is operating normally, but clocks for BISA circuits are disabled and LFSR and MISR are reset to their pre-defined state. A gating mechanism presented in Section IV-C3 will block signals from activating original circuits to OBISA circuit. Therefore, OBISA circuit stays quiet and does not consume any dynamic power. In addition, these idle OBISA cells can fulfill the role of decap cells [11] [12]. Note that OBISA circuitry will, however, consume leakage power. The other mode, *authentication mode*, is used to authenticate a fabricated chip in the field. In this mode, LFSR generates a random test pattern at every clock cycle and the test pattern is shared by all OBISA blocks. At the same time, MISR collects responses from OBISA blocks and eventually produces a signature, as shown in Figure 2(b). The test phase ends when a sufficient number of test patterns have been applied. Since inputs to OBISA come from LFSR and original circuits, thus test patterns for OBISA are generated depending on the LFSR and the state of the original circuit. We can keep the state of original circuit and run a large number of clock cycles on LFSR to test the inserted circuit as one iteration. In the next iteration, we change state of the original circuit and perform the tests once again. Test coverage will increase as more iterations are performed. Functional simulation at the OBISA design phase could help us find an efficient combination of the circuit state and the seed in the LFSR.

Note that clock is provided externally either in normal mode or in authentication mode. Typically, authentication test clock is much slower than functional clock [15]. The authentication clock frequency is dependent on the number of gates in an OBISA block and the gate types. The timing constraint can be obtained using post-layout timing analysis tool to find the longest path after OBISA inserted circuit. In the authentication mode, a very slow test clock can be used to ensure no path fails in OBISA circuits.

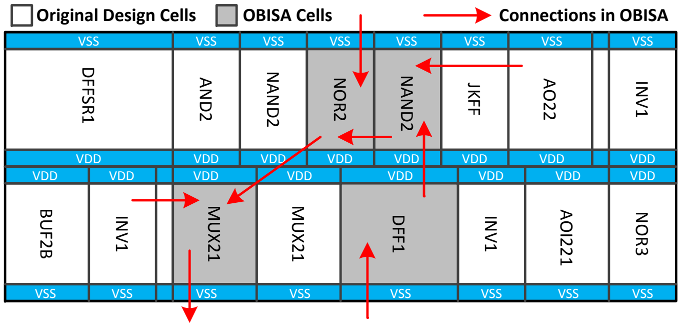
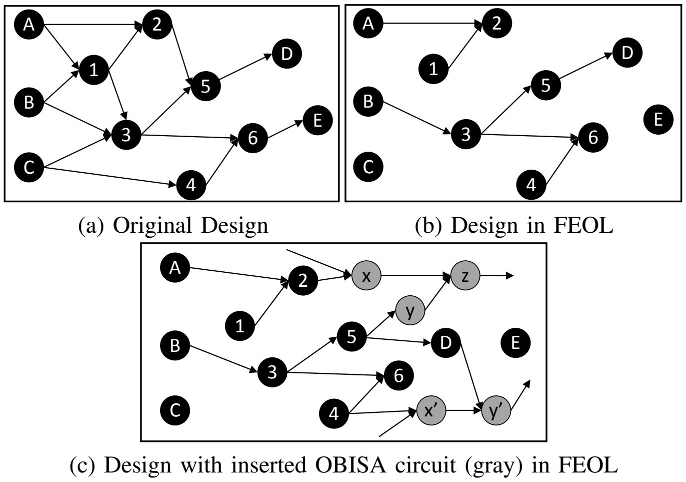


Figure 3: Layout after OBISA insertion.

C. Security Analysis

Split manufacturing obfuscates the design by preventing an untrusted foundry from gaining a full view of the layout. Our proposed OBISA structure can improve obfuscation through the following features:



(a) Original Design

(b) Design in FEOL

(c) Design with inserted OBISA circuit (gray) in FEOL

Figure 4: Circuit graphs: OBISA circuit is used to obfuscate the original design.

- Functional filler cells are inserted into unused spaces during layout design, as shown in Figure 3. Original cells and additional OBISA cells are all standard cells from the same technology library.
- Additional cells are connected together to construct combinational circuits. There is no difference in routing between original circuits and inserted OBISA circuits. While tree-structure circuit (no fan-out) is required for a BISA block in [11] [12] in order to ensure a high test coverage, fan-out is allowed in this new OBISA technique. An approach for fan-out creation in OBISA blocks is proposed in Section IV-C2.
- Interconnections between OBISA circuits and original circuits (obfuscation connections) are allowed. The additional logics can further obfuscate the original design, especially for the security-critical parts. For example, Figure 4(a) shows a design. Split fabrication can hide some wires in FEOL, but some sub-circuits (cell 3, 4, 5 and 6) could be identified. In Figure 4(c), the proposed technique can add OBISA cells to protect this sub-circuit from reverse-engineering.
- The inserted OBISA includes LFSR and MISR. We understand that LFSR and MISR have a unique structure and are controlled by mode select port, so they could be a target by an adversary to identify the additional OBISA circuit. Split manufacturing can hide critical wires in LFSR and MISR and effectively obfuscate LFSR and MISR.
- Additional local/global connections and various gates introduced by OBISA circuits can hinder neighbor connectedness analysis and standard-cell composition bias analysis [6]. OBISA blocks have not only local connections to connect cells nearby, but also long connections to other OBISA blocks, LFSR and MISR. A measure of spatial connectedness will be influenced by OBISA circuits. Similarly, additional cells with different types can change the types and proportions of cells of design presented in a small region in layout. OBISA cells can obfuscate the cell composition analysis.
- The proximity attack is based on the heuristic that floorplanning and placement (F&P) tools place the partitions close by and orient the partitions so as to reduce the wiring (delay) between the pins to be connected. [9] shows that the proximity attack could be a threat of connecting the missing nets correctly. However, OBISA circuit is able to produce additional tier-to-tier connections which can mitigate the threats of proximity attacks.

IV. Implementation Strategy

A. Implementation Flow

Figure 5 presents our proposed OBISA implementation flow. The flow fits within the conventional ASIC design flow and is computable with current commerical physical design tools. OBISA insertion procedure begins after clock tree synthesis. At that point, the whole original circuit has been placed and no more cells will be added in conventional flow (the most left column in Figure 5). The unused spaces would be identified in DEF file and various standard cells are inserted depending on size of each unused space. More information regarding OBISA cell insertion will be described in Section IV-B. Once all unused spaces are filled with OBISA cells, we will begin to connect all OBISA cells in a region to construct a number of tree-structure combinational circuits (referred to as OBISA blocks). These steps as shown in the middle column were developed in [11] [12]. The steps in the third column are proposed to strengthen obfuscation, including fanout creation in tree-structure OBISA blocks, adding obfuscation connections, and lifting secure-critical paths within OBISA. Detailed connection strategies for each of them will be presented in Section IV-C. After the OBISA process, the flow resumes the procedure in conventional design flow. The physical design tool will perform routing for the entire design including original circuit and OBISA circuit. All constraints for the original design can be taken care of by the physical design tool during routing process. Once the timing and sign-off of the design are successful, the last step involves the generation of a GDSII format of the design for final tape-out.

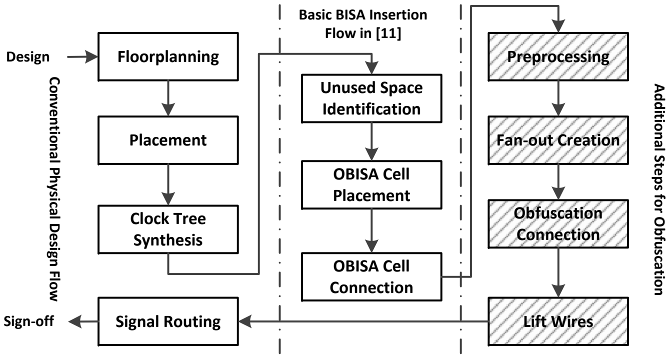


Figure 5: The OBISA implementation flow.

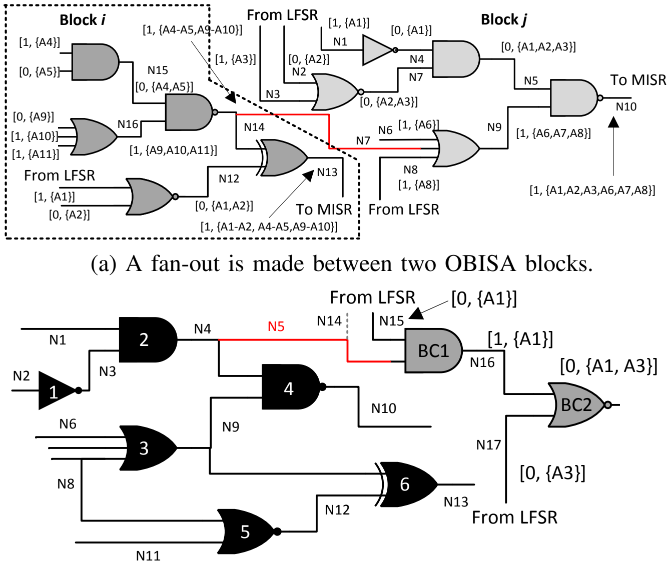
B. OBISA Cell Insertion Strategy

Several filling algorithms are proposed to fill every unused space as much as possible and no more cells could be added without changing the layout [11] [12]. Cells with different types and sizes are inserted to ensure a variety of OBISA cells due to the following reasons:

- A greater variety of OBISA cells can effectively thwart the cell composition analysis [6]. This is analogous to "dummy" cells in prior work.
- Our OBISA also supports either pre-mask or post-mask Engineering Change Order (ECO). Unused spaces are filled with a variety of standard cells and all these cells can be treated as spare logic gates. Moreover. When an OBISA cell is selected for ECO, a few modifications are needed to bypass this cell in OBISA block. Since OBISA circuits do not have a certain timing constraint, routing for OBISA could be very flexible.

C. OBISA Cell Connection Strategy

The OBISA structure allows adding fan-out and obfuscation connections. There are two points we focus on: testability of the OBISA circuit and effect on obfuscation strength. Our connection strategies are presented as follows:



(a) A fan-out is made between two OBISA blocks.

(b) An obfuscation connection is made.

Figure 6: Additional connections for improving obfuscation.

1) Preprocessing: Two pieces of information are required to collect for each net in OBISA blocks: idle state (IS) and related inputs (RI) in LFSR. As we described earlier, OBISA circuits are working in authentication mode only, so LFSR will be set to a specific state if the chip is in the other mode. We propose to set the state of LFSR to a vector that has alternative '1' and '0' to avoid biasing values in OBISA circuit. If an input is connected to the original design because of the obfuscation connection, its IS is undetermined ('X'). Therefore, the IS for each net can be '1', '0', or 'X', based on input values and their interconnections. IS can determine if a cell could be a gated cell (see Section IV-C3). Another required information is the RI for each net. Taking the OBISA circuit in Figure 6(a) for example, the RI for the net N7 are A2 (N2) and A3 (N3), because nets N2 and N3 determine the value on net N7, while the related inputs for net N5 are A1, A2, and A3. Information regarding RI can help us decide how to create fan-out without losing any test coverage (more in Section IV-C2). In this paper, we use a symbol, [x, {A,B,C}], to represent IS (x) and RI (A,B,C) of a net. One can see the IS and RI for the nets in OBISA circuits are labeled in Figure 6.

2) Fan-out Creation: Adding fan-out could potentially produce redundant gates and thereby lower controllability of gates. Fan-outs can be created by following the rules:

- The fan-out is created between a net in one block i and an input pin of another block j ($i \neq j$).
- The net in one block i and the root output in block j have no common RIs.

If these two conditions are satisfied, the net and the input pin can be a candidate pair for a fan-out. During the fan-out creation, the original connection on the input pin in block j from LFSR should be removed, so the pin is available for the new fan-out connection. These two tree-structure OBISA blocks i and j share a sub-tree circuit so that they are still tree-structure blocks in nature. Therefore, the added fan-out will not result in any extra redundant gates in OBISA blocks. Each net is still controllable, so a high test coverage of OBISA circuit will be achieved. Figure 6(a) shows an example of the fan-out creation. The net N14 in OBISA block i has completely different RI from the root output N10 in OBISA block j, so N14 can have a fan-out to connect any input in OBISA block j. In Figure 6(a), the pin for the net N7 is selected. Note that a fan-out cannot be made on the net N13, because the net N13 and N10 have shared RIs, A1 and A2.

3) Gated Cells: For logic gates, the dominant value on one input can result in a deterministic output regardless of logic values on other inputs, so those gates are gated by their dominant values. For example, '1' and '0' are used to gate OR and AND cells, respectively. We will take advantage of this for preventing dynamic power consumption within OBISA circuitry. An obfuscation connection must connect to a gated cell in OBISA circuit. For a gated cell that acts as an interface, at least one input's IS should be its dominant value during the normal operation. Other inputs of the gated cell can connect any net in the original circuit. In order to minimize modifications caused by the obfuscation connections, we prefer to choose a leaf cell for gated cell, because its input connections from LFSR can be removed without changing existing OBISA blocks. In Figure 6(b), cells BC1 and BC2 are both gated in idle state. BC1 is selected since it is a leaf cell in the tree-structure OBISA block. All gated cells will be identified at this step, and they could be selected for obfuscation connections, as described in the next subsection.

4) Obfuscation Connection: Obfuscation connection also introduces additional interconnections between OBISA circuits and original circuits. Figure 6(b) shows an example of an obfuscation connection. It will cause inevitable increased capacitance on connected nets. Although we do not worry about the timing in OBISA circuits because of the slower frequency used during the authentication for OBISA, the added capacitance could potentially fail paths in the original design. Thus, we must select connection nets very carefully to avoid timing violations. Here, we propose an approach to select nets in the original circuit for obfuscation connection based on delay estimation by the static timing analysis (STA) tool.

- We define a parameter C_0, which is a threshold for dividing paths into critical paths and non-critical paths, as shown in Figure 7.
- Given the C_0, the STA is able to find all critical paths in original design. All nets on critical paths will be excluded from obfuscation connection. All remaining nets will be assigned a virtual path delay C_0. We call it virtual path delay, because it is not a real delay. Many non-critical paths have much smaller delay than C_0, but we treat them as the same length to simplify the problem.
- Another parameter C_1 is defined as a threshold to select net for obfuscation connection. The C_1 should be smaller than functional clock period C_2. The difference of C_1 and C_2 is a safe margin that ensures no timing violation is produced due to our rough estimation.
- If an obfuscation connection is made on a net, its virtual path delay will add an increased delay D. The increased value D will vary depending on the technology libraries. It can be estimated by averaging some samples in simulations. A large enough safe margin $(C_2 - C_1)$ can tolerate such a rough calculation.
- A net can be considered for an obfuscation connection if its virtual path delay plus D is still smaller than the threshold value C_1. For example, in Figure 7, path P0s added D is still smaller than C_1. In the example of Figure 6(b), the net N4 is selected for the obfuscation connection at this step.
- If there is an available gated cell nearby (BC1 in Figure 6(b)), an obfuscation connection can be made from the net (N4) in original design to one input (N14) of the gated cell.
- The increased capacitance not only influences one net, but also affects all nets on paths that pass this net. Therefore, the virtual delay of these relevent nets will be updated by adding D. For the circuit in Figure 6(b), the net N1, N2, N3, N5, and N10 need to be updated. For the net in Figure 7, there are two obfuscation connections that push its virtual delay from P0 to P2.

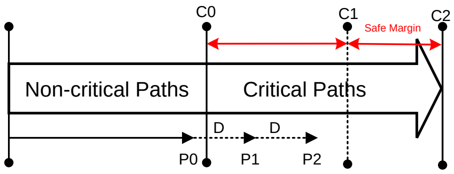


Figure 7: Net selection for obfuscation connection.

The value of increased delay D depends on the length of added wire. [18] shows that adding a short connection can bring about 30ps increased delay with 90nm technology. Since the obfuscation connection is made between nets not far away from each other, the increased delay D will not be very large. We can obtain a conservative D value in simulations.

5) Hide LFSR/MISR with Lifted Wires: Since there is one primary input that selects either function mode or test mode, an adversary could trace from this port to identify LFSR/MISR and further find OBISA cells. In order to avoid this threat, the mode select net, the feedback nets in LFSR/MISR, and some nets connecting flip-flops in LFSR/MISR should be lifted to trust tier, so attackers cannot have any opportunity to identify them.

V. Experimental Results

A. OBISA Implementation

The OBISA technique was evaluated using Opencore benchmark circuits [19]. Each circuit was synthesized with 90nm CMOS technology using Synopsys Design Compiler. Physical design, including floorplanning, placement, and routing, were conducted using Synopsys IC Compiler. Scripts were developed to analyze unused spaces in layout, insert OBISA cells, and connect them using our proposed methodologies, including fan-outs and obfuscation connections.

Table II shows the implementation results of five Opencore benchmark circuits with various scales. The number of OBISA cells for each circuit is listed in the third row of the table. Those OBISA cells do not introduce any area overhead because they are placed in the unused spaces in layout. Fan-outs and obfuscation connections are created between OBISA blocks to obfuscate their tree structures and increase the difficulty for adversaries to identify them in layout. An average 5% of nets go through trusted tier that is above M3 layer, as shown in the seventh row of the table. However, a small number of obfuscation connections are sufficient to make the inserted OBISA circuits look like a part of original design, to enhance obfuscation in FEOL. Since the test vector from LFSR has alternative '1' and '0', almost all leaf cells could be a gated cell. In our implementations, we use percentage to quantify how many inputs are altered for obfuscation connection in an OBISA circuit. Table II shows results with around 5% obfuscation connections (OCs). Thus, for an OBISA block with 80 inputs, there are 4 inputs connected to original circuit as obfuscation connections. Since all the OCs are made on short paths in the original circuit, no timing violations are introduced by OCs.

B. Authentication Test Coverage Analysis

The authentication test coverage is a metric to assess the security level of the circuit. A higher stuck-at test coverage for OBISA circuits indicates that more OBISA cells could be verified by structural test

Table II: Implementation results on different benchmark circuits.

Benchmark	DES3	USB	AES	Ethernet	DES_perf
Total Cell #	1,559	6,445	26,447	29,153	49,517
OBISA Cell #	158	439	2,950	1,169	2,090
OBISA Cell Pct (%)	10.1%	5.8%	11.1%	4%	4%
Total Net #	1,799	7,709	28,505	29,981	49,951
Secure Net # ($\geq$M4)	137	306	2,138	705	1,353
Secure Net Pct (%)	7.6%	4%	9.5%	2.4%	2.7%
Fan-out #	30	52	134	84	106
5% OC #	15	40	256	106	189

patterns. The target coverage is 100%. According to our proposed flow in Section IV-A, all inserted OBISA cells will be connected in a tree-structure manner first. Then fan-outs and obfuscation connections are added based on the existed OBISA blocks. We take the OBISA circuitry in the Ethernet benchmark circuit as a running example to compare the test coverage across different test patterns in five scenarios: *only tree-structure OBISA blocks*, *OBISA blocks with fan-out*, *BISA blocks with fan-out and three different proportions (5%, 15%, and 25%) of obfuscation connections*. For each scenario, four kinds of test patterns, 50,000, 100,000, 10,000 random patterns with different iterations, and ATPG (automatic test pattern generation) patterns, are applied to the OBISA circuits separately, and the results are shown in Figure 8. From the two left columns, we can see tree-structure OBISA circuit with and without fan-outs have the same test coverage using either random patterns or ATPG patterns. It demonstrates that the proposed fan-out creation method will not affect test coverage. The ATPG patterns can achieve almost 100% test coverage. For the random patterns from LFSR, the test coverage goes up as more patterns are applied. Their test coverage for 50,000 and 100,000 random patterns are 99.81% and 99.97%, respectively. The remaining three columns illustrate that test coverage will be impacted by obfuscation connections. This is because signals on obfuscation connections are tied to a constant value in one iteration and thus result in the controllability loss of gated cells. The last column shows an extreme scenario that 25% of inputs in OBISA blocks are connected to original design. The test coverage within 1-iteration are not high enough for the authentication test. However, as we described in Section III-B, if we change the status of original circuits with multiple iterations, i.e. the values on obfuscation connections could change, the controllability of gated cells can be compensated to some degree depending on how many iterations can perform. Results in Figure 8 show that multiple iteration offers much better test coverage than the 1-iteration with the same number of test patterns, while using 10-iteration test leads to a higher test coverage than that with 5-iteration tests. The test coverage can improve further if more random patterns are applied or more iterations are performed.

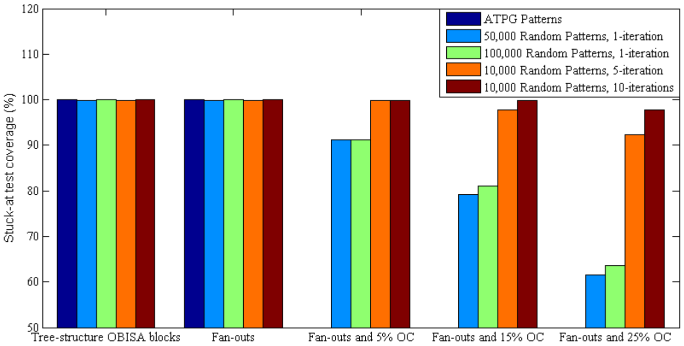


Figure 8: Authentication test coverage for an OBISA circuit.

VI. Conclusion

In this paper, we proposed a low-cost and secure OBISA technique. Split fabrication and OBISA circuit can effectively protect layout design from reverse engineering, IP piracy, and tampering. The OBISA technique fills unused spaces in layout with functional circuitry while it is also connected to the original design, in order to make the FEOL extremely difficult for an untrusted foundry to identify the added cells. The dynamic power and timing overhead is small because of net selection and gated cell selection for the obfuscation connections. The OBISA circuit can be tested in the field for hardware Trojan detection. A fan-out creation approach is proposed to obfuscate OBISA blocks further and will not affect its original high test coverage.

Acknowledgments

This work was supported in part by National Science Foundation under grant CNS 0844995.

References

[1] IARPA, “IARPA Trusted Integrated Circuits (TIC) program announcement,” http://www.fbo.gov

[2] R. Jarvis and M. McIntyre, “Split Manufacturing Method for Advanced Semiconductor Circuits,” US Patent 10/305,670, 2007.

[3] J. Valamehr, et al, “A 3-D Split Manufacturing Approach to Trustworthy System Development,” IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD), Vol. 32, No. 4, April, 2013.

[4] K. Vaidyanathan, et al, “Building Trusted ICs using Split Fabrication,” IEEE International Symposium on Hardware-Oriented Security and Trust (HOST), 2014.

[5] K. Vaidyanathan, et al, “Efficient and Secure Intellectual Property (IP) Design with Split Fabrication,” IEEE International Symposium on Hardware-Oriented Security and Trust (HOST), 2014.

[6] M. Jagasivamani, et al, “Split-Fabrication Obfuscation: Metrics and Techniques,” IEEE International Symposium on Hardware-Oriented Security and Trust (HOST), 2014.

[7] B. Hill, et al, “A Split-Foundry Asynchronous FPGA,” Proceedings of the IEEE Custom Integrated Circuits Conference (CICC), September, 2013.

[8] F. Imeson, et al, “Securing Computer Hardware Using 3D integrated Circuit (IC) Technology and Split Manufacturing for Obfuscation,” in the Proceedings of the 22nd USENIX Security Symposium, Aug., 2013.

[9] J. Rahendran, et al. “Is Split Manufacturing Secure?” IEEE Design, Automation & Test in Europe, 2013.

[10] R. Jarvis and M. McIntyre, “Split Manufacturing Method for Advanced Semiconductor Circuits,” US Patent 7195931, March 2007.

[11] K. Xiao and M. Tehranipoor, “BISA: Built-In Self-Authentication for Hardware Trojan Prevention,” IEEE International Symposium on Hardware-Oriented Security and Trust (HOST), 2013.

[12] K.Xiao, D. Forte, and M. Tehranipoor, “A Novel Built-In Self-Authentication Technique to Prevent Inserting Hardware Trojans,” IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD), Vol. 33, Issue 12, pp. 1778-1791, Dec, 2014.

[13] R. Chakraborty and S. Bhunia, “HARPOON: An Obufscation-Based SoC Design Methodology for Hardware Protection,” IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems, Vol. 28, No. 10, pp. 1493-1502, Oct, 2009.

[14] Y. Alkabani, F. Koushanfar, and M. Potkonjak, “Remote Activation of ICs for Piracy Prevention and Digital Right Management,” IEEE International Conference on Computer Aided Design (ICCAD), pp. 674-677, 2007.

[15] M. Bushnell and V. Agrawal, “Essentials of Electronic Testing for Digital, Memory & Mixed-Signal VLSI Circuits,” Springer, 2000.

[16] FreePDK45:Metal Layers. http://www.eda.ncsu.edu/wiki/FreePDK45:Metal_layers.

[17] Chipworks Inc. http://www.chipworks.com/en/technical-competitive-analysis/resources/blog/intels-14-nm-parts-are-finally-here/

[18] K. Xiao, X. Zhang, and M. Tehranipoor, “A Clock Sweeping Technique for Detecting Hardware Trojans Impacting Circuits Delay,” IEEE Design & Test, 2012.

[19] Opencores, http://opencores.org/