Logic obfuscation technique using configurable gate diffusion input for improved hardware security

Baluprithviraj Krishnaswamy Natarajan
and Vijayachitra Senniappan

Department of EIE, Kongu Engineering College,
Perundurai-638060, Erode, Tamil Nadu, India

a) baluprithviraj@gmail.com

Abstract: System-on-Chip (SoC) design using hardware Intellectual Property (IP) cores has become an integral part and pervasive practice in industries to realize error-free complex devices. However, IP vendors face major challenges in protecting hardware IPs against hardware Trojans and preventing revenue loss due to IP piracy. Obfuscation is an exact solution for protecting hardware IP against various attacks such as piracy, overbuilding and tampering. Logic locking technique allows locking outputs by fixed logic values and generates invalid output of the function if a wrong unlocking key was applied. In this paper, a novel technique called configurable Gate Diffusion Input (GDI) based logic obfuscation is proposed to enhance the security of hardware IPs. Configurable GDI based obfuscated cell inserts extra gates in the logic path of the circuit with minimum overheads to secure an IC from piracy and overbuilding. The proposed technique is simulated and synthesized using Synopsys software tool. Simulation results on ISCAS-89 benchmark circuits show that high levels of security are achieved through a well formulated obfuscation scheme at less than 10% area, power and delay overheads.

Keywords: intellectual property, obfuscation cell, gate diffusion input, reverse engineering, logic locking, physical unclonable function

Classification: Integrated circuits

References

[1] M. Rostami, et al.: “A primer on hardware security: Models, methods, and metrics,” Proc. IEEE 102 (2014) 1283 (DOI: 10.1109/JPROC.2014.2335155).
[2] R. Torrance, et al.: “The state-of-the-art in semiconductor reverse engineering,” IEEE/ACM Design Automation Conference (2011) 333 (DOI: 10.11452024724.2024805).
[3] J. Roy, et al.: “Ending piracy of integrated circuits,” IEEE Computer 43 (2010) 30 (DOI: 10.1109/MC.2010.284).
[4] J. Rajendran, et al.: “Security analysis of logic obfuscation,” ACM Design Automation (2012) 83 (DOI: 10.1145/2228360.2228377).
[5] J. Rajendran, et al.: “Fault analysis-based logic encryption,” IEEE Trans.
1 Introduction

The increasing cost of modernization and maintaining silicon foundries has caused a major paradigm change in the semiconductor business model. The semiconductor industry has been facing challenging security threats such as IP piracy, overbuilding, Reverse Engineering (RE) and hardware Trojans [1, 2]. Hence there is a basic need for a piracy-proof design flow that assists the IP vendor, the IC designer as well as the system designer to provide secured information.

Logic locking technique inserts additional gates into a circuit, locking the original design with a secret key. In addition to the original primary inputs, a locked circuit has key inputs in on-chip tamper proof memory as illustrated in Fig. 1. The additional logic may consists of NAND, NOR and XOR key gates or look-up table.
Logic locking or logic obfuscation is a technique that transforms an application or a design into one circuit that is operationally equivalent to the original circuit, but the recovery of the original circuit is significantly difficult to RE.

Fig. 1. Logic locking system

A review of the developments in logic locking has been presented in this section. Section 2 gives the overview of logic locking and Section 3 presents an explanation about GDI technique. Section 4 describes the proposed system. Section 5 deals with the experimental analysis and finally Section 6 concludes the proposed technique.

2 Overview of logic locking

This section presents the development of logic locking methods and algorithms for deciding the best locations for inserting key gates and analysis based on attack threats.

Roy et al. [3] have depicted a method called Ending Piracy of Integrated Circuits (EPIC) that protects chip foundry by automatically and uniquely locking each IC by asymmetric cryptographic techniques that require a specific external key. Rajendran et al. [4] have demonstrated an attack where the third party applies specific input patterns, observes the outputs for these patterns and deciphers the secret key. EPIC Techniques fix this vulnerability and make obfuscation truly exponential in the number of inserted keys. Zhang et al. [5] also have examined the combination of logic encryption and fault propagation analysis to widen fault analysis based logic encryption. It allows designers to block the output in a controlled manner. Yasin et al. [6] have analyzed the strengths and weaknesses of the (Boolean Satisfiability) SAT attack and presented a SAT attack resistant logic locking technique by adding only a few XOR/XNOR gates that thwart key distinguishing attacks. Baumgarten et al. [7] have explored a technique in an IC that adds reconfigurable logic barriers to split the inputs from the outputs such that the path from every input to output passes through a barrier to IC fabrication. Plaza et al. [8] have introduced a multiplexer based locking technique. It preserves test response allowing IC testing by an untrusted party before activation by correct key. Likewise Lee et al. [9] have attempted to improve the strength of the logic obfuscation by considering brute force attack strategy through the analysis of logic
cones. Y. Xie et al. [10] have developed a technique called delay locking to enhance the security of the existing logic locking techniques by locking both functionality and timing profile.

3 Gate diffusion input technique

GDI is a new technique of low-power digital combinatorial circuit design. The GDI cell relates to another form of pass transistor technology which appears equal to Complementary Metal Oxide Semiconductor (CMOS), but varies in the supply provided to the input terminals. The primary issues in the design of obfuscation cell are area, delay and power dissipation. These issues can be overcome by incorporating GDI technique. This technique allows a designer to reduce power consumption and area of digital circuits while maintaining low complexity of logic design in order to overcome the drawback of existing logic styles like CMOS [11, 12] and Pass Transistor Logic (PTL) [13, 14, 15].

GDI allows the implementation of complex logic functions using only two transistors. This technique is suitable for the design of fast and low-power circuits compared to CMOS and PTL techniques.

The GDI cell is shown in Fig. 2 from which the basic cell reminds the structure of standard CMOS inverter, but there are some important differences which is given below.

1) The GDI cell consists of three inputs: G (common gate input of nMOS and pMOS), P (input to the source/drain of pMOS), and N (input to the source/drain of nMOS).

2) Bulks of both nMOS and pMOS are connected to N or P respectively. Table I shows an input configuration of the simple GDI cell that corresponds to different boolean functions.

![Fig. 2. GDI cell](image-url)

The most essential variation between CMOS and GDI is that in GDI N, P and G terminals with a supply of ‘VDD’ could be given or can be grounded or input signal can be supplied based on the circuit to be designed, and hence efficiently reducing the number of transistors used in the case of most logic circuits (e.g. AND, OR, XOR, MUX and so on). The main advantages of GDI over conventional CMOS design are as follows: 1) lesser number of transistors results in low power dissipation and small delay. 2) Lesser number of transistors provide little area and lesser interconnect effects.
Conventionally, the IC design is without any concern of security technique to RE, piracy and overbuilding and so the aim of the proposed technique is to modify the original netlist to produce an obfuscated netlist which is functionally equivalent when the correct key is given. A logic obfuscated gate level netlist is synthesized into the layout geometry for manufacturing IC. An adversary cannot identify original functionality by image processing based RE. The modified netlist responses with Physical Unclonable Function (PUF) perform desired functions as long as correct license is issued by designers or owners.

4 Structure of proposed GDI based obfuscation cell

4.1 Obfuscation cell (OC)

Obfuscation Cell is a single GDI cell input of P (input to the source/drain of pMOS), and N (input to the source/drain of nMOS) acts as the key in this OC cell. G acts as input. The structure of GDI cell is shown in Fig. 3 where the keys K1 and K2 are used to configure GDI cell as inverter or simply a wire. The proposed technique would replace an inverter with the GDI cell or insert the GDI cell into any wire of gate level netlist. For the combination of \{K1, K2\} in the given cell, \{1 0\} act as a inverter, and \{0 1\} act as a wire. As shown in Fig. 4(a) and 4(b), a simple circuit is obfuscated by two OCs. The security of this OC is analyzed below.

![Fig. 3. GDI OC cell](image)

| N | P | G | Out | F |
|---|---|---|-----|---|
| 0 | B | A | A'B | F1 |
| B | 1 | A | A'+B | F2 |
| 1 | B | A | A+B | OR |
| B | 0 | A | AB | AND |
| C | B | A | A'B+AC | MUX |
| 0 | 1 | A | A' | NOT |

Table 1. Different boolean logic implementation using GDI

Based on the keys \{K1,K2\} and \{K3,K4\}, the OCs would act as either inverter or wire gate. The outputs of Fig. 4 would be the same only when \{K1 = 1, K2 = 0\} and \{K3 = 1, K4 = 0\}; otherwise the results may differ from the original circuit. The keys are known only to the chip manufacturer and IP/IC designer. Only the designer can authorize and guarantee the proper functionality of
the circuit. The functionality of original circuit and obfuscated circuit is given in Table II with the combination of K1, K2, K3 and K4 as \([0101], [1001], [0110]\) and \([1010]\) respectively.

![Logic circuit](image)

**Fig. 4.** (a) Logic circuit without obfuscation (b) Logic circuit with obfuscation
4.2 Impact of obfuscation technique

In the existing obfuscation technique [17, 18], a combination of multiplexer and inverter is used. By using the proposed obfuscation cell, input patterns cannot be easily identified with the stream of output patterns $O_1$ and $O_2$ with different key combinations of minimized area and overhead. The system would be complex for an attacker to predict the key and inputs thereby increase the hardware security. Similarly, more number of inverters and wires can be replaced by the obfuscation cell to increase the security of the system without the increase in the area overhead.

4.3 Test attacks and framework for security

The security of logic obfuscation and IC camouflage methods are analyzed in [9] and [10] by using the strong logic locking technique which enables structural transformations to increase the number of pair wise secure key gates. In logic obfuscation, the attacker can deduce the primary inputs by relocating the obfuscated gates based on the primary outputs Out1 and Out2. Protection scheme based on obfuscation presents antipiracy and tamper proof qualities every stage of the hardware design is proposed in [16] and [17].

The overbuilding and piracy attacks are prevented with the use of chip dependent license that can be generated by the interaction of the configuration of OCs with the PUF response [18] and the pay per device licensing service based on OC can be provided in the circuit. An attacker cannot predict the accurate license code without the knowledge of OC key to unlock the chip.

After turning on the chip, the PUF responses are XORed with the license code to get the accurate key values of OCs and the same bits are also stored in the flipflops to activate the chip. Even if the attacker extracts the gate level netlist of the obfuscated circuit by RE technique, this technique will not contain the OC keys.

Hence the attacker cannot extract the key bits and input pattern, thereby preventing the piracy and overbuilding attacks.

Due to noise and other sources, the PUF responses are not stable and the error correcting code may be used to manage such uncertainty. The chip can be unlocked only when the PUF response and OC keys generate the correct license code. Four OCs with keys $K_1$ to $K_4 = 0101$ are shown in Fig. 5. The OCs are used to replace

| in1 | in2 | in3 | out1 | out2 | OC1 AS INVERTER | OC2 AS INVERTER | BOTH OC AS INVERTER |
|-----|-----|-----|------|------|-----------------|-----------------|-------------------|
| 0   | 0   | 0   | 0    | 1    | 1               | 0               | 0                 |
| 0   | 0   | 1   | 0    | 0    | 1               | 0               | 1                 |
| 0   | 1   | 0   | 0    | 0    | 1               | 0               | 1                 |
| 0   | 1   | 1   | 0    | 1    | 1               | 0               | 1                 |
| 1   | 0   | 0   | 0    | 1    | 1               | 0               | 0                 |
| 1   | 0   | 1   | 0    | 0    | 1               | 0               | 1                 |
| 1   | 1   | 0   | 1    | 0    | 0               | 1               | 0                 |
| 1   | 1   | 1   | 1    | 1    | 0               | 1               | 0                 |

Table II. Functionality of the circuit with and without obfuscation
OR and AND gates. Various 4 bit PUF responses are generated and XORed with the key bits K1 to K4. When the XORed result is equal to 1100, considering the license code for this case as 1100, the chip gets unlocked, otherwise the chip would be in locked condition.

### 5 Experimental results

Table III shows the comparison of area, delay and power for the proposed logic with obfuscation cell using four keys and without obfuscation circuits and obfuscation circuits using two keys as mentionly Zhang [18] which is simulated and synthesized by using Synopsys simulation tool. The experiments are performed on the circuits, which are described in Verilog HDL Programming and the ISCAS benchmark circuit OCs that are used to replace the total number of wires in inverters.

The consumption of area, delay and power for bench mark circuits are observed. For fair comparison results are compared with MUX based OC insertion proposed in [18]. The performance analysis on the area, time and power consumption has been compared and shown in Fig. 6, Fig. 7 and Fig. 8 respectively. Fig. 9 and Fig. 10 shows the diagram of s420 and s349 benchmark circuits by using Synopsys software tool.

| S. No. | Benchmark | Number of inverter | Without obfuscation | With obfuscation using two keys | With obfuscation using four keys |
|-------|-----------|--------------------|---------------------|---------------------------------|---------------------------------|
|       |           |                    | Area Power Delay    | Area Power Delay                | Area Power Delay                |
| 1     | $S_{27}$ | 2                  | 25.13 7.81 0.30     | 26.12 9.31 0.41                | 25.64 8.57 0.32                |
| 2     | $S_{298}$ | 44                 | 855.56 57.71 0.13   | 864.38 64.58 0.19              | 857.13 61.85 0.14 1.98 8.87 6.25%
| 3     | $S_{344}$ | 59                 | 973.18 77.29 0.38   | 993.35 80.12 0.45              | 975.62 78.98 0.41 0.25 2.14 7.31%
| 4     | $S_{349}$ | 57                 | 969.24 76.93 0.32   | 987.23 81.32 0.43              | 972.81 78.34 0.35 0.37 1.79 8.57%
| 5     | $S_{382}$ | 59                 | 1187.96 74.22 0.12  | 1195.74 79.44 0.17             | 1189.25 77.41 0.14 0.11 4.12 7.14%
| 6     | $S_{386}$ | 41                 | 694.79 78.35 0.51   | 704.52 83.34 0.74              | 697.38 80.57 0.62 0.37 2.76 9.67%
| 7     | $S_{400}$ | 58                 | 1193.78 74.56 0.33  | 1208.17 82.28 0.19             | 1196.58 79.19 0.36 0.23 5.85 8.33%
| 8     | $S_{420}$ | 78                 | 1033.5 56.26 0.65   | 1042.6 64.57 0.83              | 1034.55 61.92 0.78 0.10 9.14 8.97%
| 9     | $S_{444}$ | 62                 | 1203.3 81.53 0.12   | 1210.4 89.23 0.20              | 1206.26 85.98 0.17 0.24 5.18 7.85%
| 10    | $S_{510}$ | 32                 | 1240.31 95.59 1.04  | 1251.6 100.5 1.16              | 1244.12 98.32 1.13 0.30 2.78 7.96%
| Avg.  |           |                    | 25.64 8.57 0.32     | 25.13 7.81 0.30                | 25.64 8.57 0.32                | 25.64 8.57 0.32                |

Fig. 5. PUF based license generation
Fig. 6. Comparison of benchmark circuits based on area consumption

Fig. 7. Comparison of benchmark circuits based on power consumption

Fig. 8. Comparison of benchmark circuits based on delay consumption

Fig. 9. Schematic diagram of s420 benchmark circuit
From the results obtained, it can be concluded that the proposed obfuscation cell using GDI technique is suitable for realizing energy efficient logic locking system.

6 Conclusion

The hardware security obfuscation technique has been reviewed to thwart piracy, overbuilding and image based RE attacks. A modified new logic style OC cell has been proposed with two key inputs. A configurable GDI technique also is proposed to improve the security of present logic locking technique. Designers can increase security level by replacing any number of inverter and wires without raise in the area, delay and power overhead. Using the proposed technique the experiment shows that area, power and delay overhead due to obfuscation are only on average of 0.41%, 4.93% and 7.9%, respectively by using Synopsys software tool which means that the proposed GDI technique can obfuscate the design without performance degradation.

Fig. 10. Schematic diagram of s349 benchmark circuit