Protecting an Intellectual Property Core during Architectural Synthesis using High-Level Transformation Based Obfuscation

Article *in* Electronics Letters · May 2017

DOI: 10.1049/el.2017.1329

Dipanjan Roy
Institute for Development & Research in Banking Technology

Protecting an Intellectual Property Core during Architectural Synthesis using High-Level Transformation Based Obfuscation

D. Roy and A. Sengupta

For protecting an Intellectual property (IP) core, it must be harder to reverse engineer. Structural obfuscation can play an important role in achieving this goal. In this letter, we propose a novel structural obfuscation methodology during architectural synthesis using multiple compiler based high level transformations (HLT) that yield functionally equivalent designs (data flow graphs) which are camouflaged in identity. The proposed obfuscation methodology is driven though a number of high level transformation techniques such as redundant operation elimination, logic transformation and tree height transformation. In addition to performing obfuscation, performing area-delay tradeoff during exploring low cost obfuscated design is also possible using these HLT techniques in the proposed methodology. Due to multiple stages of HLT incorporated in the proposed approach during obfuscation, it yields a highly robust design which on integration with particle swarm optimization based exploration framework produced low cost obfuscated IP designs. Results of proposed approach yielded an enhancement in strength of obfuscation of 20.19 % and reduction in obfuscated design cost of 59.66 % compared to a similar approach.

Introduction: With the mounting popularity of the reusable Intellectual Property cores, security threats like reverse engineering, piracy and hardware Trojan infection have become a serious problem for electronic designs. It is estimated that 10% of the globally sold electronic products are counterfeited that leads to approximately $100 billion of revenue loss [1]. Therefore major attention is required to safe-guard a reusable digital IP from an adversary. Obfuscation is a process of transforming an original design into its functionally equivalent form that significantly enhances the reverse engineering complexity [2].Though there has been prior literature which targets obfuscation-based IP core protection at lower design abstraction levels, however there is absolutely no work that provides compiler driven high-level transformation (HLT) based obfuscation for protection of reusable IP cores at architecture level. Therefore a paradigm shift in research on the protection of IP core is required to thwart reverse engineering by concealing the structure of an IP design. An obfuscation which incurs minimal design cost, provides high robustness and retains correct functionality, for obscuring the structure of a reusable IP core at architecture level is critical for the present day complex electronic designs.

As mentioned before, there exists no work on obfuscation of IP core during architectural synthesis. More explicitly, no approach in the literature has proposed a compiler based multi-stage high-level transformation driven obfuscation for robust protection of IP core at architecture level. However, few approaches such as [3] [4] have applied single-stage obfuscation for Digital Signal Processing (DSP) circuits. Unlike proposed approach, which executes obfuscation on the data flow graph (DFG) representations of reusable IP core, [3] and [4] performs on DSP circuits. Moreover, [3] and [4] (being single-stage obfuscation technique) is not as robust as proposed approach. Finally, [3] and [4] incurs higher design cost than proposed approach.

Problem Formulation: For a given data flow graph (DFG) and user provided constraints for area (A_c) and delay (L_c), explore the design space to determine a low cost obfuscated design solution during architectural synthesis. The generated solution should minimize the obfuscated design cost (shown in eqn (1) below) while satisfying user area-delay constraints; 'A' & 'L' are area and delay of an obfuscated design solution, while 'A_{max}' & 'L_{max}' indicates maximum values of area and delay of an obfuscated design solution in the design space.

$$C(obf) = \varphi_1 \frac{L-L_c}{L_{\max}} + \varphi_2 \frac{A-A_c}{A_{\max}} \qquad (1)$$

Proposed low-cost obfuscation framework: This letter proposes a novel multi-stage HLT driven obfuscation methodology during architectural synthesis. In the proposed approach, structural obfuscation is achieved through multiple stages of compiler-based HLT which includes: (a) '*Redundant Operation Elimination*' (*ROE*) (b) '*Logic Transformation*' (*LT*) (c) '*Tree Height Transformation*' (*THT*). The primary reason to employ HLTs for proposed obfuscation of reusable IP core during architectural synthesis is its ability to generate many camouflaged (but functionally equivalent) DFG designs that leads to ambiguity. The framework of proposed low-cost obfuscation methodology is shown in Fig. 1. The proposed approach accepts as inputs an original design of the application in the form of DFG, module library and user constraints (area-delay) and generates a low-cost optimized obfuscated IP design as output. The low-cost obfuscated IP is obtained by processing through the particle swarm optimization design space exploration (PSO-DSE) block [5]. The PSO-DSE (where each particle encoding indicates a resource configuration for implementing an obfuscated IP) receives the obfuscated DFG from the obfuscation block, evaluates the fitness, determines the local and global best solution and finally yields a low-cost optimized obfuscated IP design solution. The details of proposed obfuscation process are explained in the next paragraph.

One of the high-level transformations which is applied on the input DFG to obfuscate is ROE. This technique scans top to bottom of the complete DFG, identifies the redundant operations (*i.e. operations with same input & computation type, like another operation in the graph*), eliminates them and performs necessary adjustments in the graph. For example, in Fig. 2 a redundant operation is node 2 (marked as red) in the original design which is eliminated through the proposed approach to structurally obfuscate the design. To maintain the correctness of the output both the inputs of node 3 is taken from node 1 in the obfuscated design. Another high-level transformation which is applied on the DFG to obfuscate is 'logic transformation' that is responsible for modifying a DFG with some different logically equivalent function. It modifies the graph such that the graph looks obfuscated than the original yet obeys

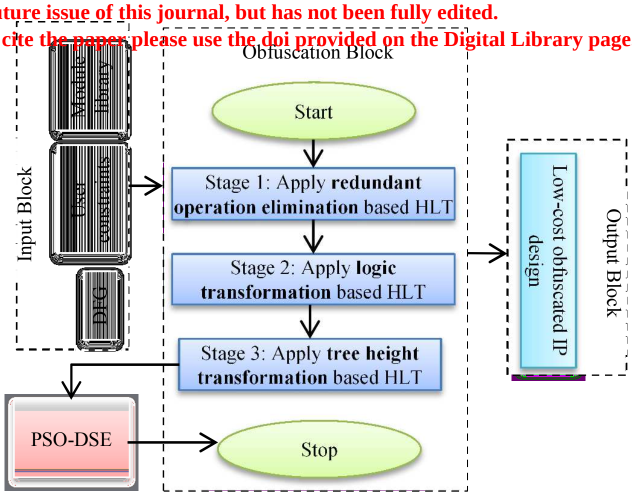
Input Block
Module library
User constraints
DFG
Obfuscation Block
Start
Stage 1: Apply redundant operation elimination based HLT
Stage 2: Apply logic transformation based HLT
Stage 3: Apply tree height transformation based HLT
PSO-DSE
Stop
Low-cost obfuscated IP design
Output Block

Fig. 1 Proposed low-cost obfuscation framework

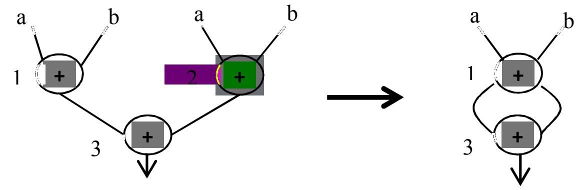

Fig. 2 Example for redundant operation elimination

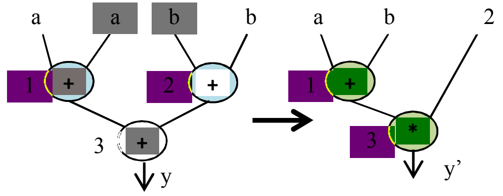

Fig. 3 Example for logic transformation

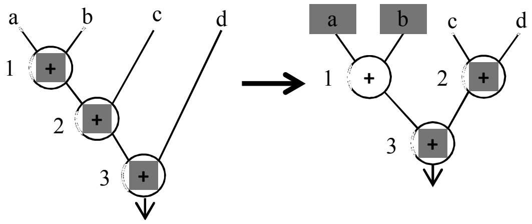

Fig. 4 Example for tree height transformation

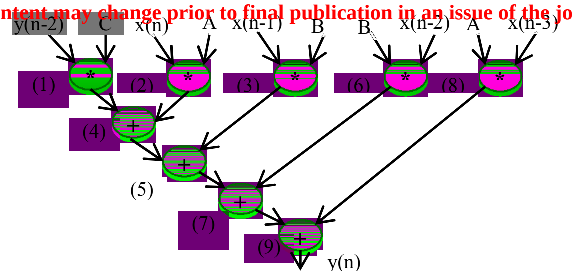

Fig. 5 Original non-obfuscated DFG of IIR filter

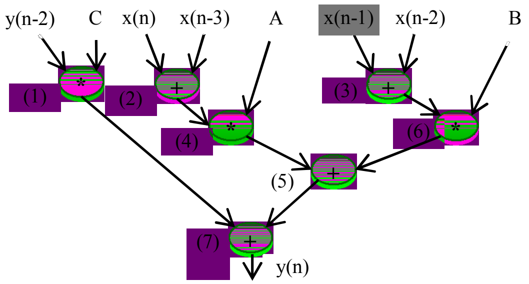

Fig. 6 Obfuscated IIR filter

the correct functionality. For example, in Fig. 3 the graph on the right is the LT-driven obfuscated form of the original graph (on the left); however both produce functionally equivalent outputs. Another high-level transformation which is applied on the DFG to obfuscate is 'tree height transformation' that is responsible for increase or decreases in height of the DFG. For example in Fig. 4, structurally obfuscated form of the original graph (on the left) is obtained by breaking the critical path dependency into temporary sub-computations which is functionally equivalent. In the proposed approach, we have performed the three aforesaid high-level transformations in successive stages (refer Fig. 1) to obtain higher robustness during obfuscation.

Motivational Example: Fig. 5 shows the original, non-obfuscated DFG of infinite impulse response (IIR) filter. Orange coloured nodes represent multiplier and purple coloured nodes represent adder in the graph. The integer value beside each node indicates the corresponding node number. As shown in Fig. 5, x(n), x(n-1), x(n-2) represents the input variables for the filter in time domain, y(n) and y(n-2) represents the current and the previous output of the filter respectively in time domain. A, B and C represents the constant values. The total number of node is 9 and the height of the tree is 5. Fig. 6 shows the functionally equivalent, structurally obfuscated design of IIR filter. All the nodes have been modified except node 1. Total number of node is reduced from 9 to 7; tree height is reduced from 5 to 4 in obfuscated design compared to the original non-obfuscated design (shown in Fig. 5). Fig. 7 shows the low-cost obfuscated IP design of IIR filter scheduled based on 2 adders and 2 multipliers (obtained through PSO-DSE block).

Experimental Result: The proposed approach is implemented in java and executed on Intel Core-i5-3210M CPU with 4GB DDR3 memory at 2.5 GHz. Design cost comparison of proposed obfuscation-based approach with approach [4] is shown in Fig.8. The result indicates that the proposed approach while abiding by user resource constraints, obtains lower obfuscated design cost (by 59.66%). On the contrary [4] incurs more hardware and latency while providing obfuscation-based protection. The proposed approach obtains lower obfuscated design cost as PSO based DSE and multiple HLTs are performed jointly. Further for proposed approach, strength of obfuscation (SoO) is higher (by 20.16 %) than [4] which indicate stronger (robust) IP protection as shown in Fig. 9. This is because proposed approach performs multi-stage HLT in succession for more camouflaging (robustness). SoO represents a normalized value between 0 to 1, determined using the following eqn.

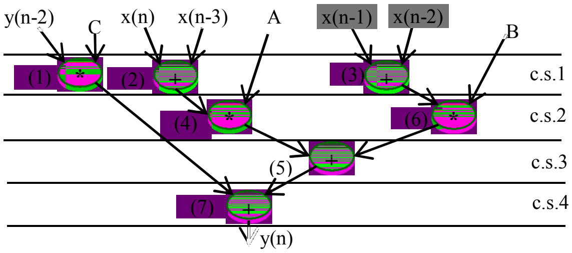

Fig. 7 Low-cost obfuscated IIR filter IP design

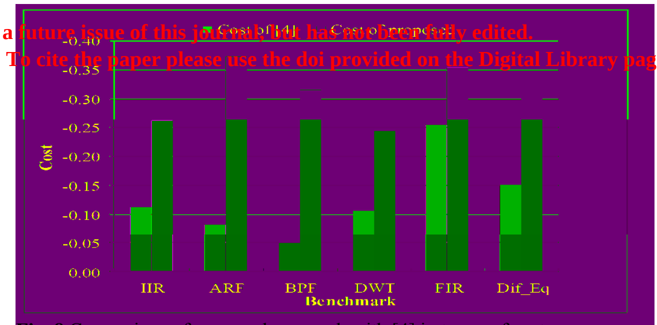

Fig. 8 Comparison of proposed approach with [4] in terms of cost

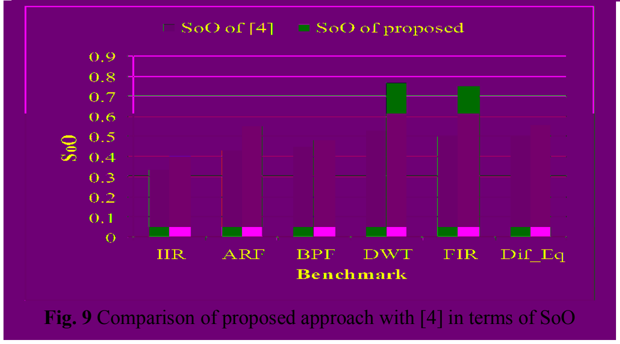

Fig. 9 Comparison of proposed approach with [4] in terms of SoO

$$\text{SoO} = \sum_i^n \frac{a_i}{a_i^T} \Big/ m \qquad (2)$$

Where a_i is the number of modified nodes due to ith HLT technique; a^T_i is the total number of nodes before applying ith HLT technique; m is the total number of HLT techniques applied on a particular application. A node is considered a modified node when either of the following is true:

- A parent node or a primary input of a node of an obfuscated DFG is different than its original.
- The child of a node in an obfuscated DFG is different than its original.
- The resource type of a node in an obfuscated DFG is changed.
- A node of original DFG is non-existent in an obfuscated DFG.

Conclusion: This letter proposes a novel multi-stage HLT driven obfuscation methodology during architectural synthesis that provides robust IP protection with low design cost.

Acknowledgments: We would like to thank Indian Institute of Technology Indore for their support in executing this research.

D. Roy and A. Sengupta (*Indian Institute of Technology Indore*, *India*)
E-mail: asengupt@iiti.ac.in

References

1 Guajardo, J., Kumar, S. S., Schrijen, G. J., and Tuyls, P., "Brand and IP protection with physical unclonable functions," *IEEE International Symposium on Circuits and Systems*, Seattle, WA, pp. 3186-3189, 2008.
2 Chakraborty, R. S., and Bhunia, S., "HARPOON: An Obfuscation-Based SoC Design Methodology for Hardware Protection," in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 28, no. 10, pp. 1493-1502, Oct. 2009.
3 Parhi, K. K., "Verifying equivalence of digital signal processing circuits," *46th Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*, Pacific Grove, CA, pp. 99-103, 2012.
4 Lao, Y., and Parhi, K. K., "Obfuscating DSP Circuits via High-Level Transformations," in *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 23, no. 5, pp. 819-830, May 2015.
5 Sengupta, A., and Bhadauria, S., "User Power-Delay Budget Driven PSO Based Design Space Exploration of Optimal k-cycle Transient Fault Secured Datapath during High Level Synthesis", *Proceedings of 16th IEEE International Symposium on Quality Electronic Design (ISQED)*, CA, pp. 289 – 292, Mar 2015.