VOLUME-8, ISSUE-2, FEBRUARY-2019 • PRINT ISSN No 2277 - 8160

Original Research Paper

Engineering

OBFUSCATION OF DSP CIRCUITS THROUGH HIGH LEVEL TRANSFORMATIONS

D.Naga Sudha Asst.prof ,JNTUHCEJ

ABSTRACT This paper gives an overall idea to design confounded circuits for digital signal processing applications by utilizing high level transformations, confounding finite state machine (FSM) depends on a key and reconfigurator. The intention is design of Digital signal processing circuits which may reused by the particular working techniques by originator. This design focused at high level transformations of repeated state graphs which have been utilized for speed, power compromises. The primary idea to evolve a design process to use high level transformations that not solely meet these tradeoffs however additionally change the architectures each structurally and functionally. Many modes of operations are introduced for obfuscation wherever the outputs are meaningful from an indication process point of view, however are functionally incorrect. Many meaningful modes are make use to reconfigure the filter order for various applications. Still existing modes may correspond with non meaningful modes. Functional obfuscation is fulfill by the right value key, and configures data. Incorrect input key is unsuccessful to change the reconfigurator and an incorrect configure data generates either a meaningful however non functional or non understandable mode. Here we have a liability to carry some chance of actuating the right mode, which ends up the decreased operations to confounded DSP circuit. The efficiency of proposed implementation is verified with IMAGE SCALING WITH INTERPOLATION AND DECIMATION design, strong high level obfuscation is proved and analyzed for various key sizes.

KEYWORDS : High level transformations,Finite state Machine, Image scaling

I. INTRODUCTION

The problem of hardware security is a serious concern that has led to a lot of work on hardware prevention of piracy and intellectual property (IP) [1], which can be broadly classified into two main categories: 1) authentication-based approach, or 2) obfuscation-based approach. Obfuscation-based approach [1] is of interest in this paper, which is a technique that transforms an application or a design into one that is functionally equivalent to the original but is significantly more difficult to reverse engineer. Some hardware protection methods are achieved by altering the human readability of the Hardware description language (HDL) code, or by encrypting the source code base cryptographic techniques. Recently, a number of hardware protection schemes have been proposed that modify the finite-state machine (FSM) representations to obfuscate the circuit's .However, to the best of our knowledge, no obfuscation based IP protection approach has been proposed for DSP circuits [1] in the literature. This paper, for the first time, presents design of obfuscated DSP circuits via high-level transformations that are harder to reverse engineer. From this standpoint of view, a DSP circuit is more secure, if it is harder for the adversary to discover its functionality. In other words, a high level of security is achieved if the functionality of a DSP circuit is designed to be hidden to the adversary our goal is to design obfuscated circuits by applying high-level transformations during the design phase. The key idea of the proposed work is to generate meaningful design variations by exploiting high-level transformations [4]. A critical challenge for nano electronic systems is to achieve yield and reliability. As VLSI technology scales into the nanometer scale, devices and interconnects are subject to increasingly prevalent defects and significant parametric variations. Based on photolithography, we are making layout features of smaller dimensions than the wavelength of the light, which requires increasingly complex OPC and other DFM techniques [3] at increasing layout area cost. Future nano electronic systems are expected to be based on self-assembly manufacture of physical structure, and achieve. Reconfiguration is further critical for nano electronic systems [5] to achieve yield and Reliability by bypassing defective or degraded devices and interconnects [4], which occurrence cannot be avoided or reduced below a certain level as is determined by the uncertainly principle of quantum physics .In this paper, we present that reconfigurable computing [2] is further a critical technology to achieve hardware security in the presence of supply chain adversaries. In recent years, a growing number of software based security solutions have been migrated to hardware-based security solutions for much enhanced resistance to software based security threats. Such systems range from smartcards to specialized secure co-processing boxes, wherein hardware provides the source of security and trust for a number of security primitives. However, in recent years, it has been brought into light that hardware is also subject to a number of security threats. The existing techniques mostly focus on information leak from a hardware system: An adversary may extract cryptographic keys and confidential information from a system by testing reverse engineering, or side-channel analysis.

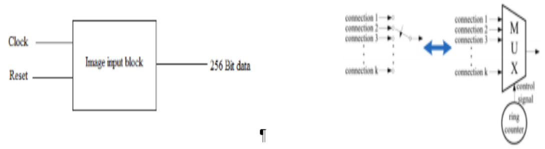
In today's global IC industry, a supply chain adversary, such as an IP provider, an IC design house, a CAD company, or a foundry may have access to the source code of the design, and may easily tamper a hardware system by planting time bombs which compromise hardware computation integrity, or creating back doors which enable information leak, bypassing access control mechanisms at higher (e.g., OS and application) levels. The recently-released Comprehensive National Cyber Security Initiative has identified this supply chain risk management problem as a top national priority A supply chain adversary's capability is rooted in his knowledge on the hardware design. Successful hardware design obfuscation would severely limit a supply chain adversary's capability if not preventing all supply chain attacks. Section II to show resent trents of hardware intellectual property (IP) piracy and reverse engineering. Section III DSP hardware protection methodology through obfuscation by hiding functionality via high level transformations. Section IV to implement simulation verified. Section V to verified different value FSM modes and reduced area.

II. HIGH LEVEL TRANSFORMATION

A supply chain adversary is an insider who is involved in the design and manufacturing of a hardware device. The tamper capability is based on his role in the supply chain, specifically, his read and write permission in the design and the manufacturing process of a specific device. An IP provider [4] or a designer for a specific module may have limited access to the design, while a foundry or a chip-level integration designer has access to the whole device design. The general lack of access control in today's supply chain further facilitates an adversary to gain knowledge of a design and launch attacks. Besides based on his role in the supply chain, a supply chain adversary may gain further knowledge of a design by probing, testing, side-channel analysis, or reverse engineering. The state-of-the-art VLSI logic encryption/locking techniques [2] include combinational logic locking and finite-state machine (FSM) locking. Combinational logic locking augments a combinational logic network [3] with an additional group of lock inputs such that the augmented combinational logic network has the same function as the original combinational logic network only if a specific vector (aka a valid key) is applied to the lock inputs .The simplest combinational logic locking technique is to insert XOR and XNOR gates into a combination logic network . An adversary knows which inputs are functional inputs and which inputs are lock inputs. He can then identify the lock gates connected to the lock inputs. If a total of M lock gates are inserted in a combinational logic network, the complexity for an adversary to find the correct logic may not be 2M. Another combinational logic locking technique is to insert multiplexers or combine logic functions based on Shannon expansion. The reason is as follows. If a lock input is connected to a lock gate that is not a XOR or XNOR gate, the key to the lock input is implied to be the non-controlling logic value of the lock gate An adversary can then easily obtain the key, unless the lock input is connected to multiple lock gates and is implied to have conflicting logic values - for example, the lock input is connected to a group of AND gates and a OR gate which have the same function as a XOR or XNOR gate. Recent trends of hardware intellectual property (IP) piracy and reverse engineering pose major business and security concerns to an IP-based system-on-chip (SoC) design flow we propose a Register Transfer Level (RTL) hardware IP protection technique based on low-overhead key-based obfuscation of control and data flow. The basic idea is to transform the RTL core into control and data flow graph(CDFG)and the integrate a well obfuscated finite state machine (FSM) of special structure, referred as" Mode-Control FSM" ,into the CDFG in a manner that normal functional behavior is enabled only after application of a specific input sequence.

III. DESIGN FLOW OF THE OBFUSCATED DSP CIRCUIT APPROACH

A novel DSP hardware protection methodology through obfuscation by hiding functionality via highlevel transformations. This approach helps the designer to protect the DSP design [5] against piracy by controlling the circuit configuration among the generated variation modes F G SR clk Reconfigurator reset re-set state M U X . . . select signal connection 1 connection 2 connection k Obfuscating configuration FSM key (switch instances).



¶

Fig1: Counter design for DSP obfuscation

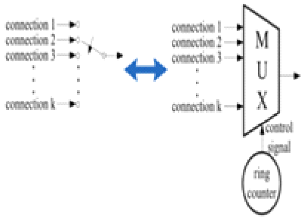


Fig2: Switch design for DSP obfuscation

'The detailed design flow is described below:

Step 1: DSP algorithm. This step generates the DSP algorithm based on the DSP application [3]

Step2: High-level transformation selection. Based on the specific application, appropriate high-level transformation should be chosen according to the performance requirement (e.g., area, speed, power or energy).

Step 3: Obfuscation via high-level transformation. Selected high-level transformations are applied simultaneously with obfuscation where variation modes, and different configurations of the switch instances are designed.

Step 4: Secure switch design. The secure switch is designed based on the variations of high-level transformations. Note that different configure data could be mapped into the same mode, which only involves simple combinational logic synthesis.

Step 5: Two-level FSM generation. The reconfigurator and the obfuscating FSM are incorporated into the DSP design as shown in Fig2. The configuration key is generated at this step.

Step 6: Design specification. This step includes the HDL and netlist generation and synthesis of the DSP system. The proposed design methodology does not require significant changes to established verification andtesting flows. In fact, the obfuscated DSP circuit with the correct key behaves just like the original circuit.

a) Secure switch design:

Here we use that the DSP circuits can be obfuscated via high-level transformations by appropriately designing the switches in a secure manner. The switches generated by high-level transformations are periodic N to 1 switches. These switches can be implemented as multiplexers, whose control signals are obtained from ring counters (as shown in Fig 3.Thus, the security of the switch relies upon design of the ring counters such that the outputs of the ring counters can be obfuscated. A ring counter is often modeled as an FSM. An FSM is usually defined by a 6-tuple (I, O, S, S0, F,G), where is a finite set of internal states, I and O represent the inputs and outputs of the FSM, respectively, F is the next-state function, G is the output function, and S0 is the initial state. However, unlike general FSMs, the FSM [2] of a ring counter is input independent, such that it always transits to the next state based on the current state. As a result, the control signal of the switches (i.e., output of the FSM) will be periodic.

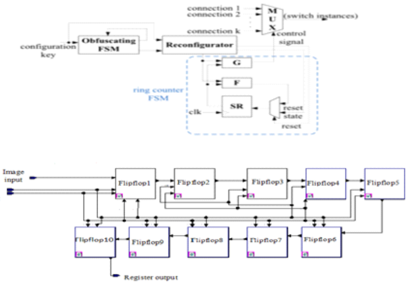


Fig3: Reconfiguration mechanism for Obfuscation.

b) Proposed Design Application:

Our scaling method requires low computational complexity and only one-line memory buffer, so it is suitable for low-cost VLSI implementation. Fig. 4 shows block diagram of the seven stage VLSI architecture for our scaling method. The architecture consists of seven main blocks: counter module (CM), register bank (RB), Interpolator and decimator and the controller. Each of them is described briefly in the following subsections

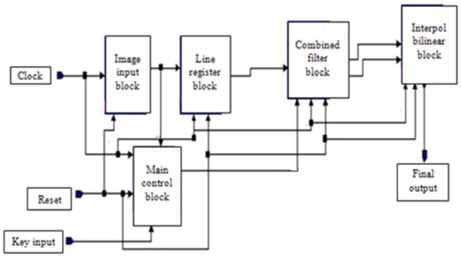


Fig4: Representing the Image Scaling application with interpolation and decimation filters

C) Algorithm & Flowchart:

1. Counter Module: This module is mainly utilized for sequence and generate image pattern for the specified application.

2. Register Bank: Here the purpose of the register bank is to provide the different shifting and storing of the image data in different

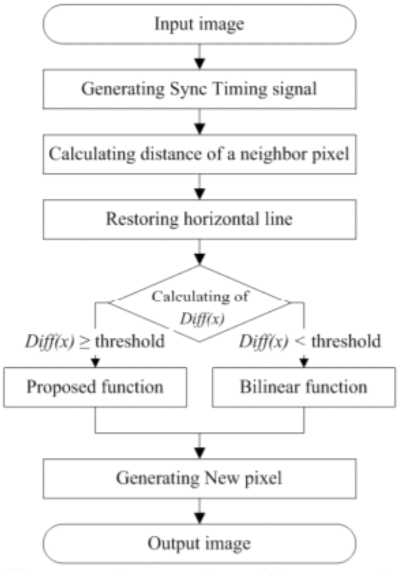
levels.

3. Interpolation: Interpolation is a method of constructing new data points within the range of a discrete set of known data points.

4. Decimation: Decimation by an integer factor, M, can be explained as a 2-step process, with an equivalent implementation that is more efficient:

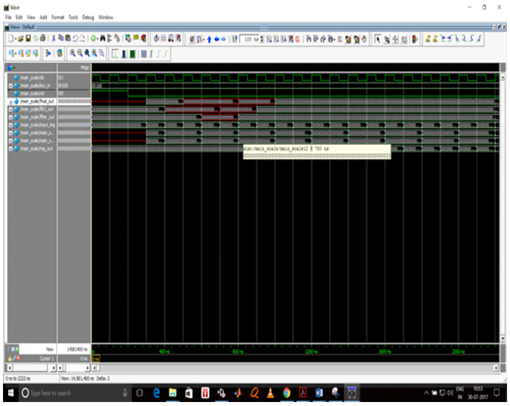
a. Reduce high-frequency signal components with a digital lowpass filter.
b. Downsample the filtered signal by M; that is, keep only every M th sample.

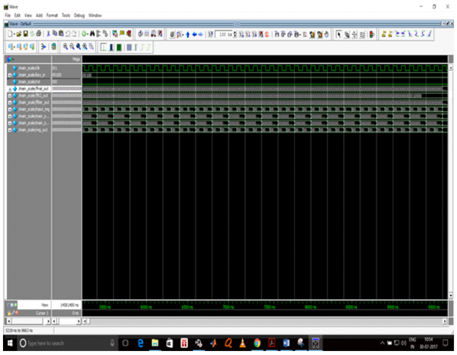
Downsampling alone causes high-frequency signal components to be misinterpreted by subsequent users of the data, which is a form of distortion called aliasing. The first step, if necessary, is to suppress aliasing to an acceptable level. In this application, the filter is called an anti-aliasing filter, and its design is discussed below. Also see undersampling for information about downsampling bandpass functions and signals.When the anti-aliasing filter is an IIR design, it relies on feedback from output to input, prior to the downsampling step. With FIR filtering, it is an easy matter to compute only every Mth output. The calculation performed by a decimating FIR filter for the nth output sample is a dot product:

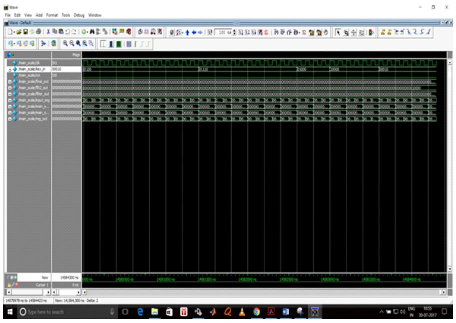


$$y[n] = \sum_{k=0}^{K-1} x[nM - k] \cdot h[k],$$

IV. SOFTWARE IMPLEMENTATION RESULTS







V. CONCULSION

This paper presents a low- overhead solution to design DSP circuit that are obfuscated both structurally and functionally by utilizing High Level Transformations techniques. It is shown that verifying the equivalence of DSP circuits by employing High Level Transformations will be harder if some switches can be designed in such a way that are difficult to trace, A securer configurable switch design is incorporated in the proposed design scheme to improve the security. A complete design flow is presented in the proposed obfuscation methodology the variation modes and the additional obfuscating circuits could also be designed systematically based on the High-Level Transformations. Obfuscated and reconfigure FSM modes of which reduce the area of performance speed improved to 341.53MHZ.

This work gives an insight into the most generalized architecture that can be customized for other image processing applications too. Though, the most fundamental point operations on the image are discussed, the idea may be carried forward for designing filtering applications also. The major challenge in this work is to choose a proper FPGA for prototyping, since the memory buffer needs enormous memory, the crucial aspect is to choose such an FPGA which has enough RAM, FIFO resources.

REFERENCES

[1]. R. S. Chakraborty and S. Bhunia, "RTL hardware IP protection using key-based control and data flow obfuscation," in Proc. 23rd Int. Conf. VLSI Design, Jan. 2010, pp. 405–410.

[2]. R. S. Chakraborty and S. Bhunia, "HARPOON: An obfuscationbasedSoC design methodology for hardware protection," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 28, no. 10, pp. 1493–1502, Oct. 2009.

[3]. R. S. Chakraborty and S. Bhunia, "Hardware protection and authentication through netlist level obfuscation," in Proc. Int. Conf. Comput.-Aided Design, Nov. 2008, pp. 674–677

[4]. W. P. Griffin, A. Raghunathan, and K. Roy, "CLIP: Circuit level IC protection through direct injection of process variations," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 20, no. 5, pp. 791–803, May 2012.

[5]. F. Koushanfar and Y. Alkabani, "Provably secure obfuscation of diverse watermarks for sequential circuits," in Proc. Int. Symp.Hardw.-Oriented Security Trust, Jun. 2010, pp. 42–47.