Journal of Electronic Testing (2022) 38:9–20
https://doi.org/10.1007/s10836-022-05984-2

Hardware Obfuscation for IP Protection of DSP Applications

Naveenkumar R[1] · N.M. Sivamangai[1] · Napolean A[1] · G. Akashraj Nissi[2]

Received: 15 November 2021 / Accepted: 28 January 2022 / Published online: 14 March 2022

Abstract
With an increasing risk of circuit piracy and intellectual property (IP), it is necessary to solve the problem of hardware security in digital signal processing (DSP) via hardware obfuscation. To obscure the circuit at a structural level, a high level of transformation techniques is used. High-level transformations (HLT) not only help in obfuscating the architecture of the circuit, it simultaneously meets the area-speed-power trade-offs. A key-based multiplexer design is proposed for the switch instance, which gives the desired output to the next node only if the configuration key is correct. A single bit change in the key will affect the whole functionality of the design. This key-based multiplexer helps to achieve functional obfuscation. As a result, two-level security is achieved. The objective of this paper is to prevent reverse engineering by structurally and functionally obfuscating the DSP circuit. Implemented and analyzed the area of the obfuscated 3-tap, 5-tap finite impulse response (FIR) filter, and obfuscated infinite impulse response (IIR) filter. Results are compared with those of the non-obfuscated filter circuit. Experimental results show that by applying the high level of transformations, the circuit gets obfuscated. Despite that, the area is reduced. The results confirm that the area of the obfuscated third-order IIR filter design is reduced by 24.56% as compared with its corresponding non-obfuscated filter.

Keywords Hardware security · Digital signal processing (DSP) · Hardware obfuscation · Intellectual property (IP) protection · High-level transformations · Structural obfuscation · Functional obfuscation · Two-level security

1 Introduction

In today's fast-growing world, technology and innovations are highly essential. Electronic circuits play a major role in this modern era. Digital signal processing circuits are a category of electronic circuits that make digital communication reliable and accurate. DSP is employed in many varieties of areas such as audio, speech processing, radar, sonar, digital image processing, seismology, the medical field, and more. Electronic gadgets are becoming increasingly interconnected with human life, safety, and piracy security has become a major challenge in this field [15]. Finite impulse response (FIR) filters are most stable as compared with Infinite Impulse Response (IIR) filters due to their non-feedback nature. But, the High-Level Transformation (HLT) method that we applied to modify the structure of the filter, employs feedback methodology. And it leads to an overall area and power consumption are reduced with an increase in security. Although the FIR filter IP core should be utilized to attenuate signals, the focus of this research is on the security of data-intensive hardware accelerators (third-party IP components of DSP). This suggested technique protects an SoC's security by providing a secure and authenticated DSP-based third-party IP component (to an SoC integrator). Using a similar procedure, other third-party IP components are normally protected before being added to an SoC. This ensures that all third-party IP components utilized in a chip are secure and authentic [28]. There has been a $100 billion revenue loss due to the lack of security of the DSP circuits [13] from the frontend to the backend, integrated circuit design has a

Communicated by C. A. Papachristou

✉ Naveenkumar R
naveentamil256@gmail.com

N.M. Sivamangai
nmsivam@gmail.com

Napolean A
nepojustin@gmail.com

G. Akashraj Nissi
nissiakash007@gmail.com

[1] Karunya Institute of Technology and Sciences, Coimbatore-641114 Tamilnadu, India

[2] Accenture, Chennai, 600063 Tamilnadu, India

number of restrictions. Hardware security is a major concern in microelectronic-systems. Hardware security is the combination of cryptographic hardware.

Moreover Hardware Trojan (HT) is a critical component of hardware security. The HT causes the integrated circuit (IC) to be maliciously modified. [18]. the hardware security concept was legally made known after the arrival of hardware Trojans and its countermeasures to lessen or avoid this kind of issue. Hardware security can be mentioned as hardware Trojans [12], cataloging, identification, and separation here the main dangers were untrusted foundries [37].

Threats to Third-Party Intellectual Property in the Supply Chain:

Generally, in hardware security, the followings are the threats in the supply chain.

1. Third Party IP (3PIP Vectors):

Anybody, unauthorized user inside the 3PIP design firm has access could sell, alter, misuse, or reverse engineer an IP during this stage because the design is open and public.

2. Inserter for SoC (System On Chip) and DFT (Design For Test):

A malevolent entity with access to unencrypted IP during the SoC or The design can also be sold, changed, or reverse engineered during the DFT insertion phase.

3. Untrustworthy factory:

Any opponent having access to the original GDSII form for the IC design could overproduce or trade it over to a third-party. In order to exploit weaknesses, they could reverse engineer the design to acquire higher-level descriptions.

4. User, supplier, and assembly:

An intruder in the assembly and supplier phase, as well as an end-user, cannot access the original design. They could, however, reverse engineer the produced IC. Despite the reverse engineering of the IC is a time-consuming and costly procedure, In recent years, better imaging and probing methods like Focused-Ion-Beam (FIB) and Scanning-Electron-Microscopy (SEM) have made it more feasible. To reverse engineer a design, an attacker must use high-resolution imaging or x-ray, delayering, and image processing for obtain the netlist from a produced IC. If an enemy is a hostile nation or a rival ill-intentioned firm, obtaining this expensive imaging equipment is possible. That's why critical designs, such as military-grade integrated circuits, must be protected against such attacks [4]. Amir et al. Discussed the fundamental concept of obfuscation and various types to protecting IP against malicious attacks, such as reverse engineering, tampering, and piracy. Generally, they developed standard hardware obfuscation open-source benchmarks (ISCAS-85 and 89) and evaluated its obfuscation metrics against various attacks [4]. Unfortunately, it is a big challenge to the safety of hardware circuits and to avoid revenue loss due to piracy. The obstructions of hardware piracy and intellectual property (IP) is divided into hardware obfuscation-based approach [35] and authentication-based approach. Ending Piracy of Integrated Circuits (EPIC) [26], Digital watermarking [10], Physical-Unclonable Functions (PUF) based-authentication technique [11], key-locking [26], and hardware metering are major authentication-based prevention approaches. EPIC was the first logical locking technique (EPIC). EPIC Place the key controlled XOR/XNOR gate on the netlist at random, and suggest a key distribution. Make use of the cryptography public-key framework [31]. The actual standard EPIC claims to be safe through argument The Quantitative Boolean Formula (QBF) is an attempt to find the actual key, Due to the alternate quantifier of the larger key, the confusion circuit of the original circuit is difficult to handle size. EPIC [26] suggested using key control to create a new locking method barriers are re-configurable logic blocks. In logic obfuscation discusses the need to make sure all paths in the circuit from input to input go through at least some critical Heuristics to develop safety [31].

The physical unclonable function (PUF) is a circuit's primitive which abstracts secret information from integrated circuits (ICs) physical properties. PUFs are based on the delay which includes multiplexer PUFs and ring oscillator PUFs [33]. Suggested using a strong PUF-based hardware obfuscation method to lock each chip individually. This proposed method is independent of other particular development powerful PUF. In addition, once PUF has been applied in the hardware security or hardware authentication circuit, the same PUF can also be applied to the design of obfuscating key generation [30]. Method of hardware metering is applied to safeguard integrated circuits (IC) intellectual property (IP) against stealing and altering runtime [3]. This hardware obfuscation focuses on which alters an application or logic behavior into something it is similar to the original although tough to reverse engineer. In literature, many hardware security techniques are met by rewriting the hardware description language (HDL) code to make it more human-readable, or cryptographic technique based encrypt

the codes. Cryptography-based encryption is the method in its machine independent of encryption and decryption [17].

Early in the design flow, obfuscating a register transfer logic (RTL) data stream might help to build in high resistance to reverse engineering effort. The obfuscation technique was implemented in an in-house high-level synthesis (HLS) system by S. A. Islam and S. Katkoori, and the obfuscated RTL designs were synthesized to gate-level with Synopsys Design compiler targeting 90 nm CMOS technology library. They show that the proposed approach obfuscates the design with an exceptionally low likelihood of reverse engineering based on testing results on four data path-heavy benchmarks. The average area, latency, and power overheads for a 32-bit obfuscation key are 2.45%, 2.65%, and 2.61%, respectively, which are reasonable. They focused on the control data flow graph (CDFG), operations on non-critical pathways are aimed to have minimal or no performance overhead. [14].

A. Sengupta et al. developed a unique strategy for protecting DSP hardware accelerators from Reverse Engineer (RE) and Trojan implantation. A key-driven structural obfuscation is used to provide security. Key-based loop unrolling, key-based partitioning, key-based redundant operation elimination, and key-based tree height modification are used in this technique to make the architecture unobvious (incomprehensible) to an intruder. At nominal design expense, their implemented perspective on DSP hardware accelerators showed a 2.3 × increase in obfuscation strength (at gate level) compared to a recent technique (showing increased security) [14]. Despite this, no common logical obfuscation approaches address the hamming distance (HD), attack resilience, area, power, or delay overhead issues. Partial Swarm Optimization (PSO) enhanced the Gate Diffusion Input (GDI) Obfuscation Cell (OC) technique for selecting the ideal site for obfuscation cell insertion while considering HD and design overhead characteristics of circuits to reduce this problem. Simulation results using ISCAS–89 benchmark circuits show that a well-formulated obfuscation method can deliver high levels of security with less than 10% area, power, and delay overheads. Designers can also boost security levels by increasing HD levels without increasing the area, latency, or power overhead [29].

Recently DSP domain there are many security issues like adding input and output to make RE of FIR filter coefficients, protecting IPs from theft, and Key initialization and data should be available for the DSP circuit to function correctly. G. Bottegal et al. recommended adding input and output noise to make reverse engineering of FIR filter coefficients more difficult for end-users [5]. To rectify the above-mentioned problem Levent Aksoy et al. implemented the decoys in the RTL to improve security [7]. A. Alaql et al. have suggested a novel hardware obfuscation method of high-level key bit fault resistance with minimal hardware overhead to protect DSP IPs from theft and RE. This method allows for a trade-off among key bit efficiency and output quality of service (QoS), as well as graceful QoS degradation when the key's BER increases. [1]. Other issue in the DSP IP are, key initialization and the data should be available for the DSP circuit to function correctly. The resultant hardware obfuscated circuit will be difficult to reverse engineer. As a result of the DSP circuit becoming highly safe then it is difficult for the opponent to determine the behavior of the circuit although if the device is physically tampered. An increase in protection is realized if the original behavior of a DSP circuit is hidden from the opponent. This shows that the DSP circuit is more secured and it will be made tough for the adversary to discover the behavior of the design. The main objective is to create a purposeful variation of a design by exploiting high-level transformations [2].

In the DSP circuits, two-level obfuscation (Structural and Functional) is mandatory, these obfuscations achieve by the HLT techniques. X. Zhuang et al. designed hardware obfuscated design of DSP circuit utilizing HLT. They implemented a circuit to obscure the program control flow at runtime while causing minimal overhead [19]. R. S. Chakraborty et al. propose a key-based multiplexer circuit for the switch instance, in the structurally obfuscated circuit, this provides functional obfuscation to the design, and as a result, two-level security is achieved [39]. Moreover, structural obfuscation is recognized by modifying the internal node circuit and state transition by giving undesired signals in the internal nodes [8].

The proposed design methodology helps to obfuscate both function and structure techniques in a third-order IIR filter. This technique helps to prevent stealing by masking thievery and preventing excessive production in DSP circuits. The content of this article is arranged as follows. Section II represents the implementation of structural level obfuscation by high-level transformations. Section III explains the design of hardware obfuscated circuits for a functional level by switch instance. Section IV renders a summary of the two levels of security and their design flow. Section V contains the results and discussion.

2 Utilization of High-level Transformations for Hardware Obfuscation

High-level transformation techniques [21] are one of the recent methods in obfuscation. Applying this technique at an algorithmic or architectural level helps in improving the DSP, image processing performance and that design is developed in VLSI technology. A good signal processing or image processing circuit design requires the perfect choice of algorithm, architecture design, implementation method, and synthesis technique. HLT techniques contribute significantly to the reduction of the overall silicon field, also support in achieving area-speed-power trade-offs. It is used in such as folding [25], unfolding [24], pipelining [23], Parallel

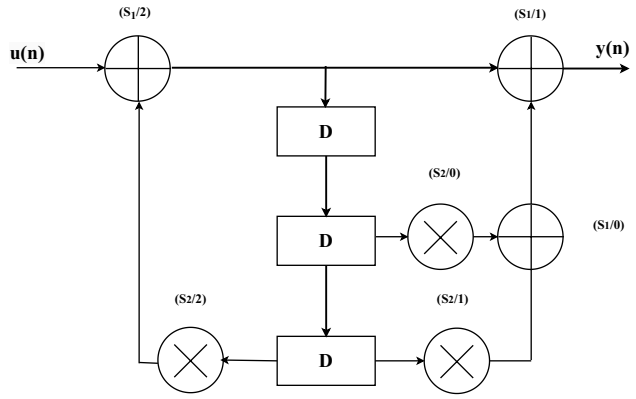


Fig. 1 3rd Order IIR Filter

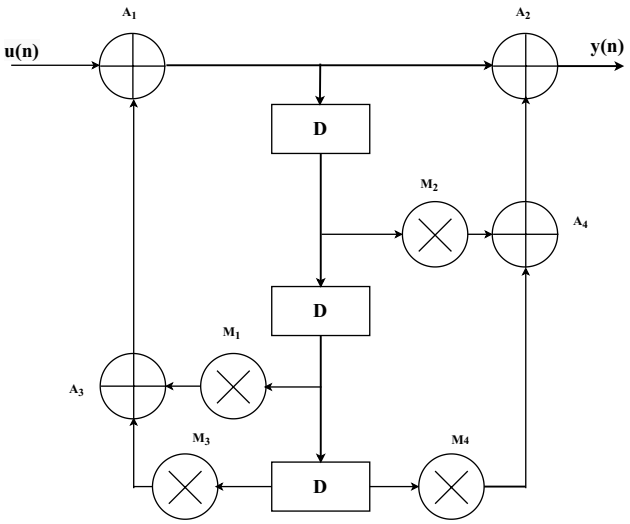


Fig. 3 Another 3rd Order IIR Filter With More Functional Units

processing [36], retiming [38], look-ahead transformations [20, 22], relaxed look-ahead [32], reduction in strength and has been used in the efficient synthesis of DSP Systems. It structurally modifies the signal processing circuit without affecting its original functionality.

Folding is an example of such transformation, through which hardware obfuscation can be achieved. Different folding sets in folding transformations develop a different folding architecture, this leads to structural obfuscation without affecting its functionality. As a result, a DSP circuit structure is obfuscated [16], which makes it difficult for the adversary to find its behavior, thereby making it hard for reverse engineering. For example, by using folding transformations, several filters could be combined into a single multiply-accumulator (MAC) [6]. Changing the structure of the designed circuit, the structure obfuscation is modified.

In this paper, structural modification of a digital signal processing design is done by applying a high level of transformations. This technique deals with structural modification in a DSP system. Directly, It does not affect the behavior of the design, it is said to be passive. These transformations focused on reducing the area with decent speed and power dissipation. By applying these techniques in the design architecture, in addition to the obfuscation, it gives as area-speed-power trade-offs. Under high-level transformation, the folding technique is an essential solution for better obfuscation. Its main purpose is to reduce the number of functional blocks in a DSP architecture. It's a concept that is contradictory to the unfolding technique. As it was mentioned before, different folding sets in folding transformations lead to different folding architecture. The order of operation is represented by the folding sets. The formulae for folding is,

$$DF(U \rightarrow V) = Nw(e) - PU + v - u$$

where,

N – Folding Factor

DF – denotes Delay element number of required between U and V.

w (e) – the amount of delays employed in the transition from U to V

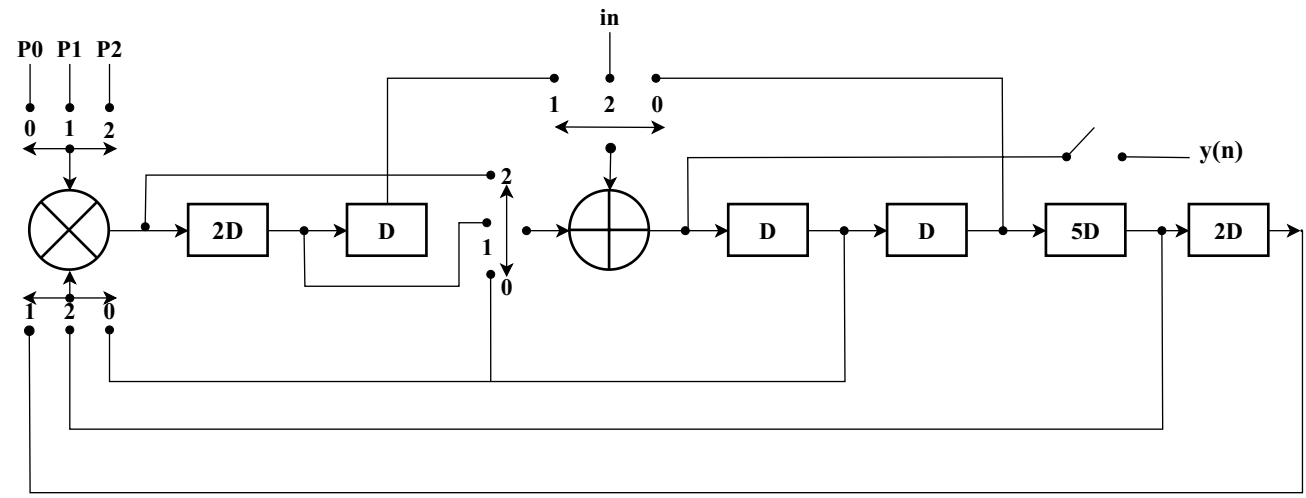


Fig. 2 Obfuscated Structure of 3rd Order IIR Filter from Fig. 1

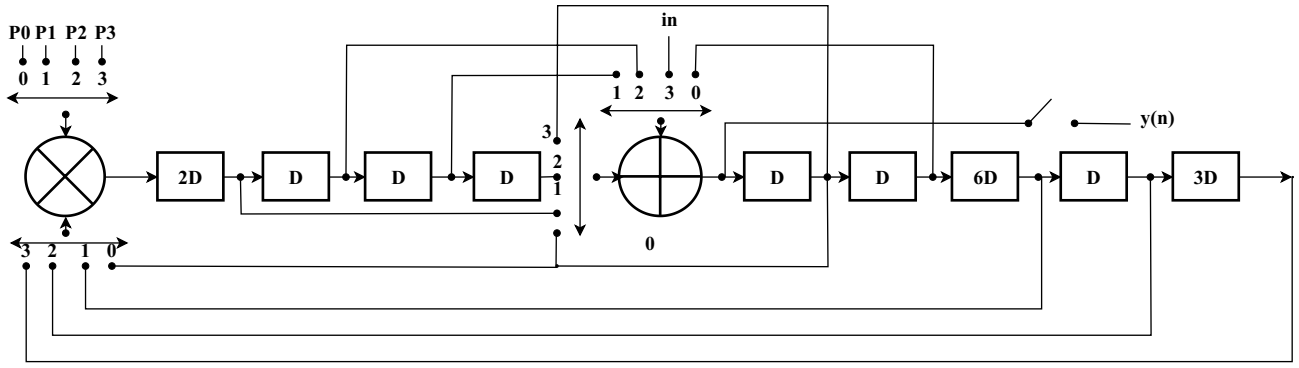


Fig. 4 Obfuscated Structure of 3rd Order IIR Filter from Fig. 3

PU − standsforɛinternaldelayɛinUschangedoperation.
uandv − UandVfoldingorders

Above this equation gives us the number of delays elements to be introduced between a specific operation cycles. It should be greater or equal to '0'.

Figure 1 shows an example 3rd order IIR filter. It has 3 adders, 3 multipliers, and 3 delay elements. Figure 2 shows the obfuscated design of the 3rd order IIR filter from Fig. 2, in that we can see that the number of functional units was reduced to 1 each and the number of delay elements has increased.

This shows that the overall area of the design has been reduced at the same time the design has been obfuscated. The switch instance plays a major role in this design. The switch instance design should be accurate for the overall operation to be meaningful. In this obfuscated design the latency is high compared to the 3rd order IIR filter's traditional design. But, the critical path delay stays almost the same.

Taking Fig. 3, it's another example IIR filter with more functional units. We can note that in Figs. 2 and 4, both the obfuscated designs appear almost the same in the architectural level except for the number of delay elements and the design of switch instance. Even an obfuscated design of an 18th order or more appears the same in architectural level as that of the obfuscated design shown in Figs. 2 and 4, only the number of elements of delay and the design of switch instance differs. This shows the ambiguity in design. So, it's understood that it'll be hard to find the functionality from an obfuscated DSP circuit and it will be hard for the adversary to reverse engineer a particular design. Figures 5 and 6 show the 3-Tap FIR filter and Obfuscated structure of the 3-Tap FIR filter. Moreover, Figs. 7 and 8 show the 5-Tap FIR filter and obfuscated structure of the 5-Tap FIR filter.

3 Design of Obfuscated Switch Instance

In section II, it can be observed that high-level transformations are responsible for structural obfuscation of the signal processing circuit. Already it will be hard to find the functionality from the structurally obfuscated design, designing the switch instance can make the design even more secured. Usually, a normal multiplexer design will be used for switch instances. This paper presents a key-based multiplexer design which makes the design even more complicated and this design of switch instance obfuscates the functionality of the overall circuit. In this key-based multiplexer design, we can have a key configuration of N number of bits. The configuration key varies from design to design. Figures 9, 10 and 11 shows the idea of the design for the switch instance.

In this design, if the configuration key is correct then the data from the different nodes will reach the next node without

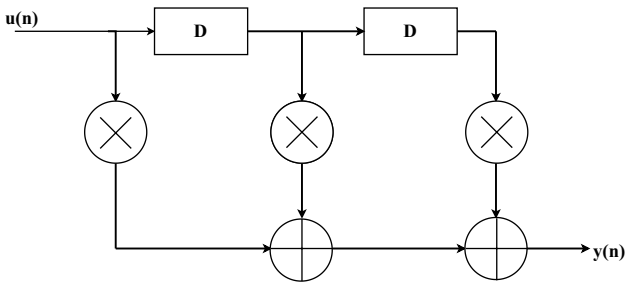


Fig. 5 3-TAP FIR Filter

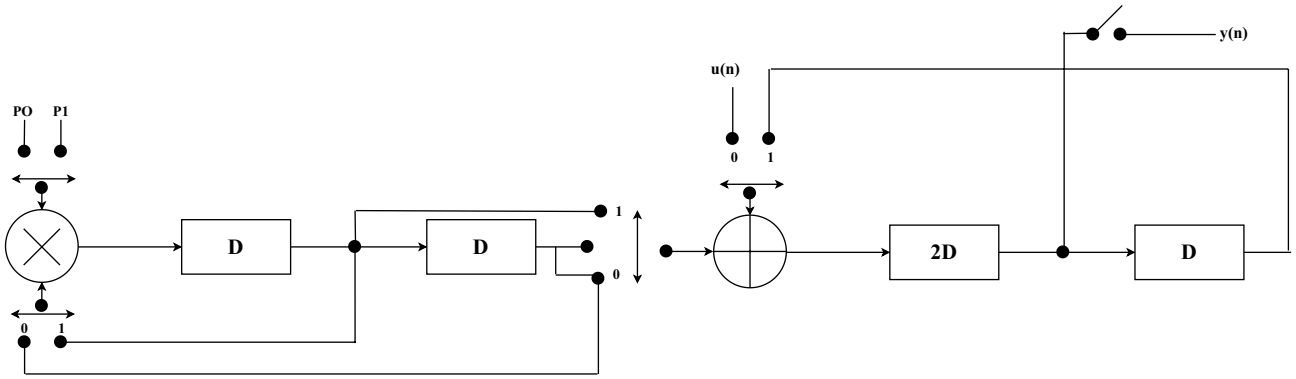


Fig. 6 Obfuscated Structure of 3-Tap FIR Filter

any confusion and the outputs will be meaningful. But if the configuration key is wrong then it produced random sequence numbers from Linear–Feedback-Shift-Register (LFSR) will be sent to the next node then the obtained output will be non-meaningful or wrong.

If the configuration key is 32-bits, then 2^{32} possible combinations should be tried out to find the correct operation. From the key configuration bit length, based on the multiplexer design's select line requirement the bits are partitioned from any part of the key, which is going to be ambiguous. This design process is done at the HDL level. And based on the instance, the counter design will help to allot the partitioned key parts to the multiplexer's selection. If the given key is wrong or even if it is in the wrong order, or even a single '1' or '0' change will give out a random number from the LFSR to the next operating node which will affect the overall working of the obfuscated design which results in wrong output. This process helps us achieve functional obfuscation. There is not going to be a single switch instance in a folded DSP circuit, there will be two or more. We can uniquely design each switch instance with a different key configuration or even a different bit length for the key configuration. One switch can have 12-bits and another can have 16-bits. Designing the circuit in this way will result in a higher degree of obfuscation.

Only 8 to 1 multiplexer does not need to be used for switch instances as shown in Fig. 9. We can use any design based on our obfuscation requirements. Figures 10 and 11 show a design for efficient synthesis technique for 10 to 1 multiplexer and 11 to 1 multiplexer respectively.

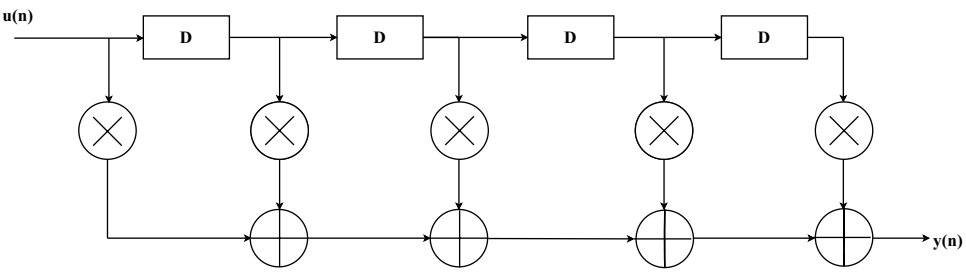
The proposed design, considering the above technique to reduce the area of design with a higher degree of obfuscation since the normal multiplexer structure changes from the conventional design.

4 Two-level Security and Its Design Flow

The main objective of this idea is to make sure that the designer's Intellectual property (IP) is not being stolen and reverse engineer is prevented. To see it from a hacker's point of view, he either tries to analyze the netlist's structure in order to identify, differentiate the actual circuit from the obfuscated one or an intruder might try to reverse engineer based on simulation, thereby determining the functionality of the design.

Structural obfuscation through high-level transformations helps to overcome the first threat of structural analysis. Through visual inspection and structural analysis, the adversary will verify the Register Transfer Logic (RTL) structure, gate-level structure, and layouts. This is a weak attack because it'll be very difficult for the adversary to find out the original behavior for obfuscated design of a large circuit of DSP. High-level transformations provide hardware obfuscations at a gate-level netlist or a HDL level. Different designs of DSP circuits can have the same obfuscated structure, just

Fig. 7 5-Tap FIR Filter



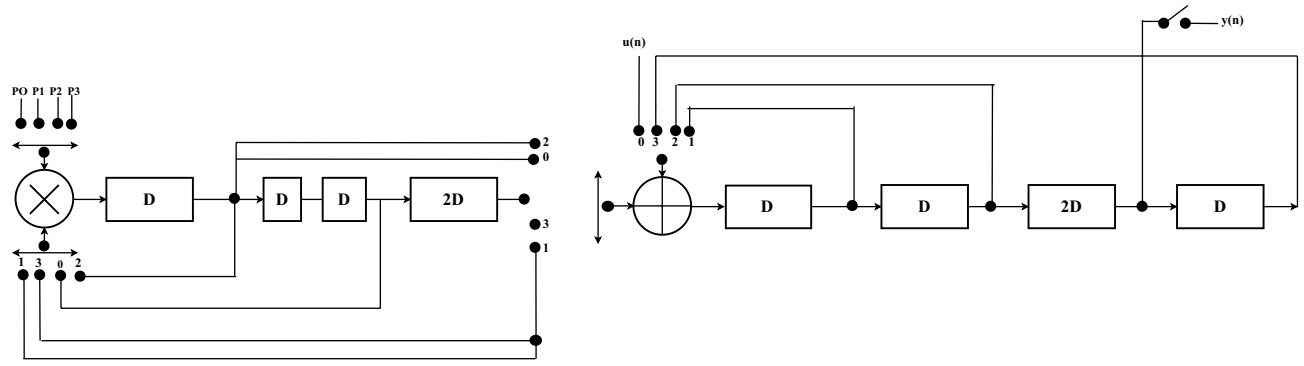


Fig. 8 Obfuscated Structure of 5-Tap FIR Filter

by altering the switch instance the entire working conditions may vary. For example, the structure of a 4th order IIR (Infinite impulse response) filter and the structure of an 18th order IIR filter will be similar, only the switch instance and the number of delay elements differ. This proves the ambiguity in the obfuscated design. If the adversary knows only of the structural information but does not know of switch instance after it will be difficult to get the logic of the original DSP circuit.

Considering the simulation-based approach, a key-based multiplexer is designed. For the obfuscated circuit to work correctly the configuration key should be correct. The configuration key varies from design to design. The number of bits for the key configuration is not fixed or specific. It can be of any number based on our design requirement. For example, if we use a 16-bit key configuration, it can be used for any order filter, that is, it can be used for 4th order as well as 16th order. The key is partitioned and based on the instance or count value it will be sent to a particular select line and gives out the meaningful output for that time instance. If the given key is wrong, then the output is going to be a random number which will affect the functionality of the whole design. So during simulation, it is going to be difficult to get the behavior of the design without knowing the exact configuration key. Even if a single bit is changed, the whole functionality gets affected. It will be hard to try out all the possible configurations. For example, if a 32-bit configuration key is used, there are 2^{32} possible configuration keys, even in these there are possibilities that the first cycle of operation works perfectly but all the other cycles have the wrong key or the first two cycles might be right and others are wrong. To improve the obfuscation efficiency, instead of using the first few bits of the configuration key in a cycle, the middle apart and last part of keys are preferred. Anyway, it depends on the number of bit

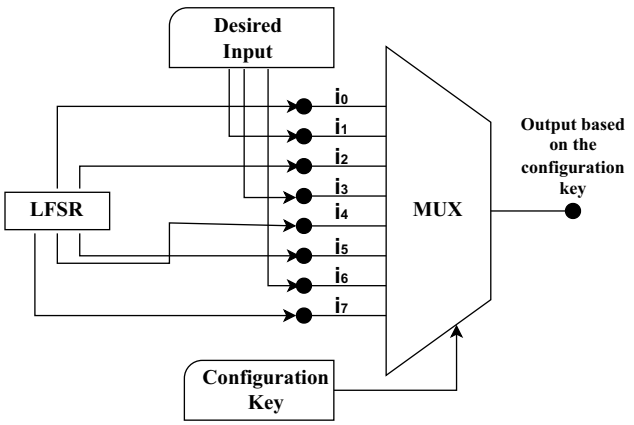


Fig. 9 Key-Based Multiplexer Design

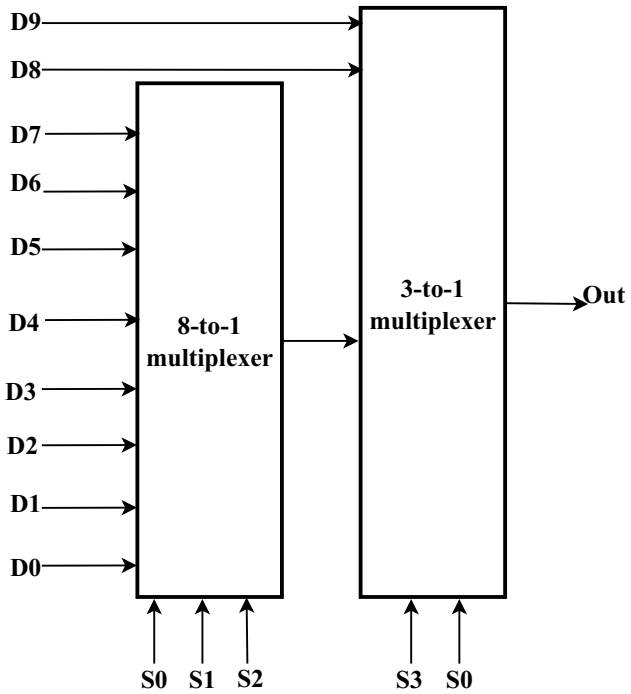


Fig. 10 Proposed 10 to 1 Multiplexer

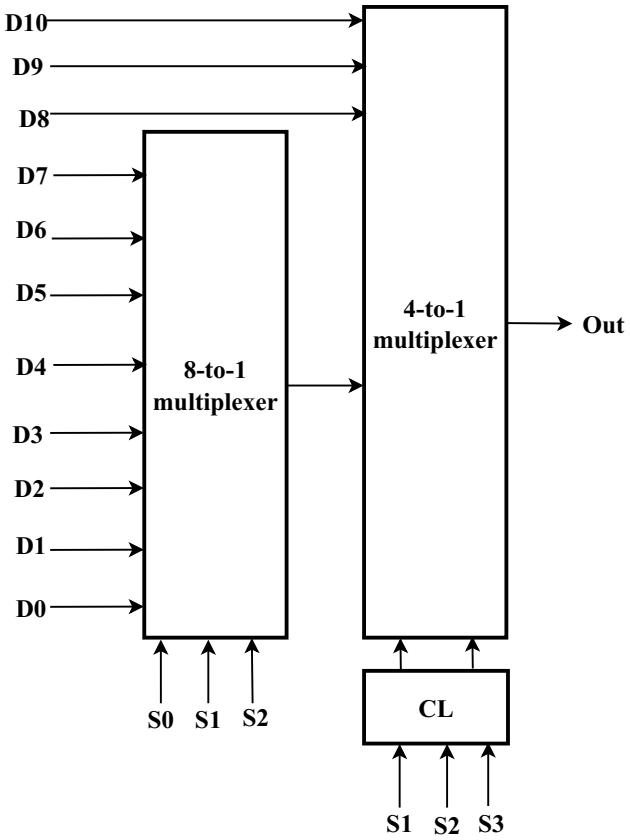


Fig. 11 Proposed 11 to 1 Multiplexer

configurations for example in 32-bit configuration 2^{32} possibilities keys are used.

There might be more switch instance in a DSP circuit and each switch instance can have a unique design which makes it even more complicated and make it difficult for the adversary to find the behavior of logic from the above explanations it can be concluded that this method for obfuscating the design considers the possible threats a design can face and shows how the levels of security can prevent the design from being stolen. And thus, the main objective is satisfied. The above methods, confirm that the designer's IP will be protected and reverse engineering will be prevented to an agreeable extent.

A. Design Flow

We put forward a DSP circuit protection approach through hardware obfuscation by hiding behaviors of the design through the transformation of high-level methods. This method supports the circuit designer for securing the digital signal processing circuits from stealing. The complete flow of the design is given here.

Step-1: Circuit design: The first step involves, the design of the DSP filter circuit depending on the DSP-based requirements of the application.

Step-2: Selection of HLT: Depending on the precise applications and performance requirements like the power, speed, and area, based the suitable High-level transformation (HLT) of technique is chosen.

Step-3: Structural obfuscation: In this step, by using the desired high-level transformation technique, the circuit is modified at a structural level. This results in a structurally obfuscated circuit which makes it hard to reverse engineer thus making that tough to identify the behavior of the logic.

Step-4: Key-based Multiplexer design: A key-based multiplexer is incorporated in the switch instance. The expected output is obtained, only if the correct key is given to the multiplexer. The wrong key, on the other hand, leads to unexpected outputs. The configuration key depends on the design of the multiplexer.

Step-5: Functional obfuscation: Functional obfuscation can be achieved with the help of the key-based multiplexer. The LFSR (Linear-Feedback- Shift- Register) is applied for creating random numbers which are given as inputs to the wrong keys. The wrong key affects the original functionality of the design and produces unexpected results.

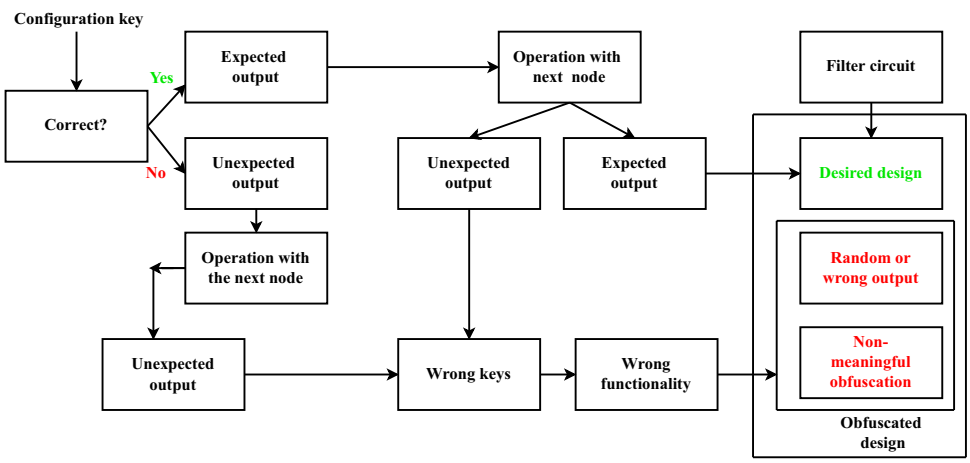


Fig. 12 Architecture of the Proposed Obfuscated Design

Step-6: Design specifications: In this step, Hardware Description Language (HDL) code for the design is written, the netlist is generated and the design is synthesized.

After, these design stages, the obfuscated circuit is sent to the DSP system manufacturing stage, the conceptual design methodology makes it impossible for the producer to gain accessibility through functioning or the Keys configuration. This provides very little information to the manufacturer. With the appropriate key, the obfuscated DSP circuits work just like the equivalent circuit.

B. Architecture

The proposed obscured DSP Architecture is shown in Fig. 12. Initialization of the key is given to the key-based multiplexer which is integrated with the switch instance. The multiplexer will send the expected output for the next operation only if the given key is correct. If the given key is wrong, the output from the switch instance will be a generation sequence of random numbers from the Linear-Feedback-Shift-Register (LFSR), which when operated with the next node gives non-meaningful results. Operations performed in the next node can produce either an expected output or unexpected output based on the results of the previous node. This process continues throughout the design and if the final output is the expected output, the desired design is obtained. On the other hand, if the final output is unexpected, random, or wrong, then it portrays a non-meaningful obfuscation. Thus this obfuscated hardware architecture helps to increase the security of the circuit and prevents piracy.

5 Results and Discussions

This paper presents an ideal solution for the hardware obfuscating DSP design both structurally and functionally using high-level transformations. The 180 nm process technology and the NC Launch and RTL compiler (Cadence) EDA tools are used. The hardware obfuscated structure is difficult for reverse engineers because of the two-level security as explained in section IV. In addition to the structural obfuscations, functional obfuscations are also achieved by introducing a key-based multiplexer in the switch instances, which makes the circuit even more complicated when compared to the normal switches. As from Table 1, it can be seen that there is a 22.11% decrease in the area when compared to 3 TAP FIR and obfuscated 3TAP FIR filter, where for 5-TAP FIR filter and obfuscated 5-TAP FIR filter there is a 43.62% decrease in area, whereas for IIR filter and obfuscated IIR filter there is 24.56% decrease in area, from which it concluded that not only the structure is obfuscated also the area occupied and the number of gates employed are also minimized, which reduces power dissipation.

In [15] the author claims that by using high-level transformations, a key-based obfuscating finite-state machine, and the area overhead of a (3 l) th-order IIR filter benchmark is only 17.7%. (FSM), but in our proposed design the area is reduced by 24.56% for the same (3 l)th-order IIR filter without comprising in security. The switch instance is designed in such a way that it doesn't take much of the area and at the same time gives us a considerable amount of security. As mentioned in the section "design of obfuscated switch instance", for a 32-bit configuration key there are 2^{32} possible combinations which are hard to try out everything to make it work as desired and the LFSR circuit make the switch design even more secured, a wrong key can change the whole functionality of the design since the random value will be pushed to the next node. In [9] the author claims that high-level hardware obfuscation is achieved with 10% of area overhead under delay constraints and In [19] the author claims that high levels of security within the under delay constraint, less than 5% area and power overhead can be achieved by his proposed work, whereas in our work we managed to reduce the area at the same time achieving equal amount of hardware obfuscation. Thus the overall area is

Table 1 Area Comparison

DSP CIRCUITS	AREA(mm^2)	NO.OF GATES
3-TAP FIR FILTER	3785	1378
OBFUSCATED 3-TAP FIR FILTER	2948	846
5-TAP FIR FILTER	6460	2344
OBFUSCATED 5-TAP FIR FILTER	3642	946
IIR FILTER	5275	1924
OBFUSCATED IIR FILTER	3979	1053

Table 2 Comparison of Key Size versus Obfuscation Metric:

S.No	MUX	L	K	Probability of being attacked	
				Proposed Work	[6]
1	8:1	4	4	$5.89*10^{-3}$	$3.90*10^{-3}$
2	12:1	8	4	$8.3*10^{-5}$	$2.44*10^{-4}$
3	16:1	12	4	$2.28*10^{-5}$	$1.52*10^{-5}$

*In the above table LFSR (Linear-feedback shift register) is excluded

Table 3 Metrics Analysis of Key Based Multiplexer

Circuit	Power (nW)	Area(μm^2)	Number of Gates	Delay (pS)
Key based Mux	109511.695	2666	490	559

Table 4 Performance Comparison of Obfuscated and Non-Obfuscated IIR Filter

Filter	Power (nW)	Area(μm^2)	Number of Gates	Delay (pS)
Non-obfuscated IIR Filter	10710571.819	46003	1924	1963
Obfuscated IIR Filter	16517449.990	34700	1053	1968
Percentage Difference	35.15% increase	24.5% decrease	45.27% decrease	0.25% increase

reduced and at the same time a considerable amount of security has been achieved and the probability [34] of the DSP circuit getting attacked is reduced. The result confirms, the equality of the DSP circuit If the switch is likely to be used, it will be more difficult to use HLT technology Made a path that is not convenient to track, The configurable switch circuit is included with expected design schemes to develop the safety [27].

Generally, the intruders use a trial and error method to find the correct key. For example, for a length (L) and Configuration key (K), the probability of being attacked is $1/2^{L+K}$. The following Table 2 shows the relation between K, L, and the Probability of being attacked. Also, it is compared with [34]. Increasing K or L values the probability of being attacked can be minimized.

So especially when we are using 12:1 mux, the probability of being attacked is reduced compared with [34]. And increasing the level of security and hard to reverse engineer. The following updated Tables 3 and 4 represent the Metrics analysis of key-based multiplexer and obfuscated and non-obfuscated IIR filter respectively.

Table 3 shows the analysis of key-based multiplexer, which is used for the obfuscated filter to make the filter hard to reverse engineer.

Table 4 shows the percentage difference between IIR Filter and obfuscated filter. It is clear from the tabulation that there is an increase in power and delay of the obfuscated filter when compared to IIR filter whereas, area and number of gates used are reduced in comparison with IIR filter.

6 Conclusion

The main idea of this work is to achieve a two-level security in the DSP hardware circuit for the designer's IP is not being stolen and making the DSP circuit hard to reverse engineer. High-level transformations play a critical role in reducing the overall silicon area. As stated in section III, it will be hard to find the functionality from the structurally obfuscated design, designing the switch instance can make the design even more secured which is accomplished. Implemented and analysed the area of the obfuscated 3-tap, 5 -tap FIR filter, obfuscated IIR filter and its results are compared with non-obfuscated filters circuit. Result proves that the amount of area reduced despite the design of several switch instances in a single design. Notably, in the obfuscated third-order IIR filter design in the area and number of gates are reduced by 24.56% and 45.27% as compared with its corresponding non-obfuscated filter. Thus the transformation of high level is yields the high-level security and down the area scaling.

Funding Nil.

Data Availability Not Applicable.

Declarations

Research Involving Human and Animal Participants Not Applicable.

Conflict of Interest Nil.

References

1. Aksoy L et al (2021) "High-level Intellectual Property Obfuscation via Decoy Constants," Proc. IEEE 27th International Symposium on On-Line Testing and Robust System Design (IOLTS). 1–7. https://doi.org/10.1109/IOLTS52814.2021.9486714
2. Alaql A, Hoque T, Forte D, Bhunia S (2019) "Quality Obfuscation for Error-Tolerant and Adaptive Hardware IP Protection," Proc. IEEE 37th VLSI Test Symposium (VTS). 1–6. https://doi.org/10.1109/VTS.2019.8758637
3. Alkabani Y, Koushanfar F (2007) "Active Hardware Metering for Intellectual Property Protection and Security." USENIX Security Symposium 291–306
4. Amir S, Shakya B, Xiaolin Xu, Jin Y, Bhunia S, Tehranipoor MM, Forte D (2018) Development and Evaluation of Hardware Obfuscation Benchmarks. Journal of Hardware and Systems Security 2:142–161. https://doi.org/10.1007/s41635-018-0036-3
5. Baluprithviraj KN, Vijayachitra S (2020) Optimization of Logic Obfuscation Technique for Hardware Security. Int J Sci Technol Res 9:1044–1048
6. Basiri MA, Sk NM (2015) "Configurable Folded IIR Filter Design," in IEEE Transactions on Circuits and Systems II: Express Briefs. 62(12);1144–1148. https://doi.org/10.1109/TCSII.2015.2468917
7. Bottegal G, Farokhi F, Shames I (2017) "Preserving Privacy of Finite Impulse Response Systems," in IEEE Control Systems Letters. 1(1):128–133. https://doi.org/10.1109/LCSYS.2017.2709621
8. Chakraborty RS, Bhunia S (2008) "Hardware protection and authentication through netlist level obfuscation," in Proc. IEEE/ACM International Conference on Computer-Aided Design. 674–677. https://doi.org/10.1109/ICCAD.2008.4681649

9. Chakraborty RS, Bhunia S (2009) “HARPOON: An obfuscation based SoC design methodology for hardware protection”. IEEE Trans Computer-Aided Design of Integrated Circuits and Sys 28(10):1493–1502. https://doi.org/10.1109/TCAD.2009.2028166
10. Chang C-H, Cui A (2010) Synthesis-for-Testability Watermarking for Field Authentication of VLSI Intellectual Property. IEEE Trans Circuits Syst I Regul Pap 57:1618–1630. https://doi.org/10.1109/TCSI.2009.2035415
11. Chang CH, Zheng Y, Zhang L (2017) “A Retrospective and a Look Forward: Fifteen Years of Physical Unclonable Function Advancement.” IEEE Circuits and Systems Magazine 17:32–62. https://doi.org/10.1109/MCAS.2017.2713305
12. Dupuis S, Flottes ML, Di Natale G, Rouzeyre B (2018) "Protection against Hardware Trojans with Logic Testing: Proposed Solutions and Challenges Ahead," in IEEE Design & Test. 35(2):73–90. https://doi.org/10.1109/MDAT.2017.2766170
13. Guajardo J, Kumar SS, Schrijen GJ, Tuyls P (2008) "Brand and IP protection with physical unclonable functions," Proc. IEEE International Symposium on Circuits and Systems, 3186–3189. https://doi.org/10.1109/ISCAS.2008.4542135
14. Islam SA, Katkoori S (2018) "High-level synthesis of key based obfuscated RTL data paths". Proc. 19th Int Sym Quality Electr Design (ISQED) 407–412. https://doi.org/10.1109/ISQED.2018.8357321
15. Lao Y, Parhi KK (2014) "Protecting DSP circuits through obfuscation," Proc.2014 IEEE International Symposium on Circuits and Systems (ISCAS), 798–801. https://doi.org/10.1109/ISCAS.2014.6865256
16. Lao Y, Parhi KK (2015) "Obfuscating DSP Circuits via High-Level Transformations," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems 23(5):819–830. https://doi.org/10.1109/TVLSI.2014.2323976
17. Methodology for protection and Licensing of HDL IP by Tarun Batra, Cadence Design Systems, Inc. Noida, India
18. Naveenkumar R, Sivamangai N. M., Napolean A. and Janani V (2021) "A Survey on Recent Detection Methods of the Hardware Trojans," Proc. 3rd International Conference on Signal Processing and Communication (ICPSC), 139–143. https://doi.org/10.1109/ICSPC51351.2021.9451682
19. Parhi KK (1989) “Algorithm transformation techniques for concurrent processors,” Proc. IEEE. 77(12);1879–1895. https://doi.org/10.1109/5.48830
20. Parhi KK (1991) "Pipelining in algorithms with quantizer loops," in IEEE Transactions on Circuits and Systems. 38(7):745–754. https://doi.org/10.1109/31.135746
21. Parhi KK (1995) High-level algorithm and architecture transformations for DSP synthesis. J VLSI Sig Proc 9:121–143. https://doi.org/10.1007/BF02406474
22. Parhi KK (2005) "Design of multigigabit multiplexer-loop-based decision feedback equalizers," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems. 13(4):489–493. https://doi.org/10.1109/TVLSI.2004.842935
23. Parhi KK, Messerschmitt DG (1989) “Pipeline interleaving and parallelism in recursive digital filters. I. Pipelining using scattered look-ahead and decomposition,” IEEE Transactions on Acoustics, Speech, and Signal Processing. 37(7):1099–1117. https://doi.org/10.1109/29.32286
24. Parhi KK, Messerschmitt DG(1991) “Static rate-optimal scheduling of iterative data-flow programs via optimum unfolding,” IEEE Transactions on Computers. 40(2):178–195. https://doi.org/10.1109/12.73588
25. Parhi KK, Wang CY, Brown AP (1992) “Synthesis of control circuits in folded pipelined DSP architectures”. IEEE J Solid-State Circuits 27(1):29–43. https://doi.org/10.1109/4.109555
26. Roy JA, Koushanfar F, Markov IL (2008) "EPIC: Ending Piracy of Integrated Circuits," Proc. Design, Automation and Test in Europe. 1069–1074. https://doi.org/10.1109/DATE.2008.4484823
27. Sandeep P, Mennaiah Batta P, Shiva Rama Krishna P, Kiran Kumar D (2020) "Obfuscation Mechanism for DSP Protection”. Int J Eng Res Technol (IJERT) 9(5):6–11. https://doi.org/10.17577/IJERTV9IS050050
28. Sengupta A, Rathor M (2020) "Enhanced Security of DSP Circuits Using Multi-Key Based Structural Obfuscation and Physical-Level Watermarking for Consumer Electronics Systems," in IEEE Transactions on Consumer Electronics, 66(2): 163–172. https://doi.org/10.1109/TCE.2020.2972808
29. Sengupta A, Rathor M, Patil S, Harishchandra NG (2020) "Securing Hardware Accelerators Using Multi-Key Based Structural Obfuscation," in IEEE Letters of the Computer Society. 3(1);21–24. https://doi.org/10.1109/LOCS.2020.2984747
30. Shahed QM, Enamul and John A. Chandy, (2019) Key Generation for Hardware Obfuscation Using Strong PUFs. Cryptography 3(3):17. https://doi.org/10.3390/cryptography3030017
31. Shamsi K, Li M, Plaks K, Fazzari S, Pan DZ, Jin Y (2019) “IP Protection and Supply Chain Security through Logic Obfuscation.” ACM Trans Design Automation of Electronic Sys 24(6):1–36. https://doi.org/10.1145/3342099
32. Shanbhag NR, Parhi KK (1993) "Relaxed look-ahead pipelined LMS adaptive filters and their application to ADPCM coder," in IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing. 40(12):753–766. https://doi.org/10.1109/82.260240
33. Suh GE, Devadas S (2007) “Physical Unclonable Functions for Device Authentication and Secret Key Generation.” Proc. 44th ACM/IEEE Design Automation Conf 9–14. https://doi.org/10.1145/1278480.1278484
34. Sunumol KS, Shanu N (2015) "Obfuscation in DSP algorithms using high level transformations for hardware protection". Proc. IEEE Recent Advs Intelligent Computational Sys (RAICS) 27–32. https://doi.org/10.1109/RAICS.2015.7488383
35. Vijayakumar A, Patil VC, Holcomb DE, Paar C, Kundu S (2017) "Physical Design Obfuscation of Hardware: A Comprehensive Investigation of Device and Logic-Level Techniques," in IEEE Transactions on Information Forensics and Security. 12(1): 64–77. https://doi.org/10.1109/TIFS.2016.2601067
36. Wu W, Wang J, Li W, Zhang W (2009) "Design Methods of Multi-DSP Parallel Processing System," Proc. WRI World Congress on Comp Sci Information Eng. 458–464. https://doi.org/10.1109/CSIE.2009.40
37. Yier J (2015) “Introduction to Hardware Security.” Electronics 4:763–784. https://doi.org/10.3390/electronics4040763
38. Zhu X, Basten T, Geilen M, Stuijk S (2012) "Efficient Retiming of Multirate DSP Algorithms," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. 31(6);831–844. https://doi.org/10.1109/TCAD.2011.2182352
39. Zhuang X, Hsien-Hsin TZ, Lee S, Pande S (2004) “Hardware assisted control flow obfuscation for embedded processors,” in Proc. International Conference on Compilers, Architecture, And Synthesis for Embedded Systems. 292–302. https://doi.org/10.1145/1023833.1023873

Naveenkumar R Karunya Institute of Technology and Sciences. Email: naveentamil256@gmail.com. Naveenkumar R, is a research scholar at the department of Electronics and Communication Engineering of Karunya Institute of Technology and Sciences, Tamilnadu, India. He got his M.E degree from Anna University, Chennai, India in 2014. He has 5 years of academic experience and 1 year of industry experience. His research interest are hardware security in microelectronics.

Dr. N. M. Sivamangai Karunya Institute of Technology and Sciences. Email: nmsivam@gmail.com. Dr. N. M. Sivamangai is an Associate Professor, Department of ECE, Karunya Institute of Technology and Sciences, India. She received her Ph.D. degree from Anna University, Chennai, India in 2011. She has 16 years of teaching experience. She was instrumental in the fabrication of IC jointly with Indian Institute of Science - Bangalore, in the year 2008. Her research interests are to design and test high performance semiconductor memories and to design VLSI based systems.

Napolean A Karunya Institute of Technology and Sciences. Email:nepojustin@gmail.com. A. Napolean is a research scholar in the department of Electronics and Communication Engineering of Karunya Institute of Technology and Sciences, Tamilnadu, India. He got his M.Tech. degree from Vellore Institute of Technology, India in 2008. He has 14 years academic experience. His research interests include advanced memory technologies, Nano electronic device fabrication, modelling and sensors system.

G Akashraj Nissi received her Bachelor of Technology degree in Electronics and Communication Engineering form Karunya Institute of Technology and Sciences, Coimbatore, India in 2018. He is currently Senior Software Engineer, Accenture, Chennai. His research interest includes Hardware security, Digital signal processing, Internet of things, Artificial Intelligence and parallel processing architecture.