B.TECH PROJECT REPORT

on

Protecting Trojan Secured DSP cores against IP Piracy

BY

Abhinav Reddy and Yash Bontala

DISCIPLINE OF COMPUTER SCIENCE AND ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY INDORE

November 2022

Protecting Trojan Secured DSP cores against IP Piracy

A PROJECT REPORT

Submitted in partial fulfillment of the
requirements for the award of the degrees

of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE ENGINEERING

Submitted By:
Abhinav Reddy and Yash Bontala

Guided by:
Dr. Anirban Sengupta

INDIAN INSTITUTE OF TECHNOLOGY INDORE
November 2022

CANDIDATE'S DECLARATION

We hereby declare that the project entitled **Protecting Trojan Secured DSP cores against IP Piracy** submitted in partial fulfillment for the award of the degree of Bachelor of Technology in 'Computer Science and Engineering' completed under the supervision of **Dr. Anirban Sengupta, Associate Professor, Computer Science and Engineering, IIT Indore** is an authentic work.

Further, I declare that I have not submitted this work for the award of any other degree elsewhere.

Signature and name of the student(s) with date

CERTIFICATE by BTP Guide(s)

It is certified that the above statement made by the student is correct to the best of my knowledge.

Dr. Anirban Sengupta, Associate Professor, CSE
Nov 30, 2022

Signature of BTP Guide(s) with dates and their designation

Preface

This report on "Protecting Trojan Secured DSP cores against IP Piracy" is prepared under the guidance of Dr. Anirban Sengupta.

Through this report, we aim at explaining our work done on developing a Trojan secured digital signal processing (DSP) cores against the threat of intellectual property (IP) counterfeiting/piracy using facial biometric based security approach . We present our motivation and approach towards the problem, the concepts used in developing the algorithm and the results obtained.

We have tried to the best of our abilities and knowledge to explain the content in a comprehensive manner. We have also added diagrams, tables and charts wherever required.

Abhinav Reddy Asireddy
Yash Bontala
B.Tech. IV Year
Discipline of Computer Science and Engineering
IIT, Indore

Acknowledgements

We wish to thank Dr. Anirban Sengupta for his kind support and valuable guidance during the course of the project. We express our heartfelt gratitude towards him for giving us an opportunity to work on the project. He has motivated us constantly to work harder towards the objectives and explore more.

It was his help and support, because of which we have been able to come up with the algorithm and the technical report.

We would also like to thank Mr. Rahul Chaurasia , Phd scholar in the discipline of Computer Science and Engineering at IIT Indore for his help and suggestions whenever we required. Working on this project was a wonderful learning experience for us and we thank everyone who motivated us in the process.

Abhinav Reddy Asireddy
Yash Bontala
B.Tech. IV Year
Discipline of Computer Science and Engineering
IIT, Indore

Abstract

This work presents a system architecture for securing the Trojan secured digital signal processing (DSP) core against the threat of IP counterfeiting/piracy using face biometric based security approach. In the implemented architecture, the Trojan secure DSP design is first built by constructing a sister unit of the original DSP design using the same mask set. Then, the facial signature is generated from the face features of genuine IP owner. Finally, this generated signature is embedded into the Trojan secure DSP core in the form of covert security constraints during high- level synthesis (HLS) phase of VLSI design process. Therefore, the proposed approach ensures stronger security of the Trojan secure DSP cores against IP counterfeiting. The security strength of the proposed methodology can be quantified using probability of coincidence (Pc) and the probability of an attacker extracting the exact signature combination to authenticate the counterfeit IP ownership "P(f)." Furthermore, the methodology incurres no design overhead while ensuring robust protection of the Trojan secure DSP cores using face biometric security.

Contents

Chapter 1

Introduction

1.1 Motivation

Digital signal processing (DSP), motion picture expert group (MPEG), Discrete cosine transform (DCT), Finite impulse response (FIR) and digital filters, such as finite impulse response (FIR), are widely used in several electronic gadgets such as digital cameras, smart phones, and portable media players. These DSP cores perform video processing tasks, such as image compression, noise filtration, and object detection. With the advancement of technology, the integration of such DSP cores into digital electronics systems accelerates their performance and efficacy. However, they are exposed to several security risks[1][2]. In today's advanced technologies such as Internet of Things(IoT), Artificial Intelligence(AI), and machine learning, the integration of such untrustable third-party IP cores into any digital devices may lead to various security issues, safety and reliability hazards to the end users. Such counterfeit designs may contain malicious hidden codes. These malicious codes may trigger the excessive heat dissipation, drain of rapid power, malfunctioning, and may also lead to performance degradation of the device. Therefore, security against counterfeit plays an important role in consumer electronics device. Moreover, the designs are said to be secure against Trojan insertion, if it is capable to detect the Trojan inserted in such designs. The presented Trojan secured design architecture enables the Trojan detection during Trojan detection process.
The proposed Trojan secured architecture facilitates the detection of Trojan present, during Trojan detection stage itself. The implemented architecture employs the distinct multi-vendor strategy used for allocating the resource in the designed which facilitates the easy detection of Trojan, if presented in the designed. But these Trojan secured designs are vulnerable to IP pirating

and ownership abuse. Therefore, along with the Trojan detection mechanism, the DSP designs needs to be secured against the threats of IP piracy/ counterfeits.

1.2 Related work

1.2.1 Hardware Watermarking

The process of implanting covert marks as design characteristics within hardware of IP design is known as hardware watermarking. During architectural synthesis, register allocation is discovered utilising the notion of a coloured interval graph, where the nodes of the graph indicate storage variables and the edges denote the existence of overlapping lifetime between variables. A coloured interval graph with an edge between two storage variables indicates that a shared register cannot be assigned to store the two storage variables. Watermarking restrictions are implemented in the HLS register allocation stage by introducing extra constraints in the form of additional edges between the nodes of a coloured interval graph. In the register allocation stage, any number of extra edges (additional restrictions) can be inserted. The bigger the number of edges (constraints), the more secure the signature. The addition of these extra edges as watermarking requirements indicates that the storage variables of a coloured interval graph are compelled to execute through different registers. Any watermarking technique must have a signature detection procedure[5].

1.2.2 Hardware Steganography

Approaches to hardware steganography produce stego-constraints that are embedded them in the form of the author's secret information into the target hardware design. Secret design data, stego-encoder,secret stego-keys, and mapping rules are used to construct .These restrictions are implemented in the HLS register allocation stage by introducing extra constraints in the form of additional edges between the nodes of a coloured interval graph. In the register allocation stage, any number of extra edges (additional restrictions) can be inserted. The bigger the number of edges (constraints), the more secure the signature. The addition of these extra edges as watermarking requirements indicates that the storage variables of a coloured interval graph are compelled to execute through different registers[6].

1.2.3 Digital Signature

Digital signature is inserted in DSP core in DFG (Data Flow Graph) based on user specification. LIST scheduling technique is used to schedule the DFG based on the supplied resource configuration. The planned DFG is now used as input to the following block. In this block, the scheduling information is retrieved and translated into bitstream using an encoding rule. The bitstream is then supplied as an input to the SHA module, which generates a DSP core bitstream digest. Before the output bitstream is encrypted in the RSA module, it is handled in postprocessing to make it compliant with the input type needed by the RSA algorithm. .Following that, the decimal output values from the prior phase are encrypted with a vendor-supplied RSA private (secret) key. This encrypted-hashed bitstream produced by the proposed method is referred to as a 'digital signature.' Each bit of the bitstream is encoded using the encoding technique given by the encoding block to incorporate this digital signature in the form of design constraints. Finally, the digital signature constraints are placed into the signature embedding block's register allocation step of high level synthesis (through the generation of a coloured interval graph (CIG)) to get the watermarked DSP core as the principal result[7].

1.2.4 Fingerprint Biometric

In this technique a digital template is generated from IP vendor's figerprint biometric data. Following that, the digital template is embedded in the hardware accelerator as hidden biometric constraints.

As a result, a secure hardware accelerator is obtained. A vendor's biometric (fingerprint) information built in the design ensures a robust technique to refute an adversary's bogus claim of IP ownership. Furthermore, by adopting a highly authentic secret mark based on an actual biometric fingerprint, the identification of IP counterfeiting/piracy will be enabled.

1.3 Overall Objectives

- Design a trojan secured DSP core that embeds system designer's facial signature during register allocation phase of HLS to protect the reusable IP core.
- Produce a solution which is trojan secured so that the consumer electronics don't lead to performance degradation.
- Ensuring stronger security in terms of detecting counterfeited designs and also incurring almost zero design cost overhead.

Chapter 2

PRELIMINARY CONCEPTS

2.1 High Level Synthesis (HLS)

High Level Synthesis is an automated design technique that takes description of an algorithm of a desired behaviour and turns it into digital hardware. From behavioural criteria and design limitations, it develops a register transfer level design. Typical HLS components include behavioural specification, dataflow analysis, operation scheduling, and register allocation. The typical steps of high level synthesis are as follows :-

1. Compilation of specified input
2. Generation of intermediate form
3. Selection
4. Allocation
5. Scheduling
6. Binding
7. Architecture generation

In context of the project, the input is a text file in the following format :-

```
*,0,0,1
*,0,0,2
*,0,0,3
*,0,0,4
+,1,2,5
+,5,3,6
+,6,4,7
```

This represents a Data flow graph (DFG), it connects operations and their

dependencies over each other. (i1,i2,i3,i4) represents an individual operation where:-

- i1 = Nature of the operation (adder / multiplier) e.g. * : multiplier, + : adder
- i2,i3 = Inputs of the operation
- d = Operation number / identifier

2.2 LIST Scheduling

List scheduling works by attempting to plan the "greatest" amount of operations in the control step while keeping resource limits and data dependencies in mind i.e, with minimum latency and resource constraints(MLRC). During the scheduling process, a ready list is employed to keep track of data-ready operations that are data dependent.. The ready list in a control step contains unscheduled activities that can be scheduled into the current control step without breaking the data dependency (i.e., actions whose immediate predecessors have been scheduled into previous control steps). As long as there are operations in the ready list that fulfil the resource limitations, they are selected and scheduled into the current control phase. As long as there are operations in the ready list that fulfil the resource limitations, they are selected and scheduled into the current control phase. If more than one operation from the ready list may be scheduled in a control step but does not meet resource limitations, a priority function chooses amongst the ready operations. "ALAPi - ASAPi" is a typical priority function (i.e., range where the operation can be scheduled). Operations with narrower ranges (i.e., lower mobility) are prioritised since there are fewer feasible control steps into which those operations can be scheduled, and deferring them to a later control step would likely increase the overall length of the schedule.List scheduling helps to reduce overall design delay by scheduling key activities early. This is a crucial distinguishing characteristic from the linked work, as scheduling in the related work is inspired by the fingerprint, whereas scheduling in the implemented approach is an independent process.

2.3 Hardware Trojan

A hardware trojan is a hardware security threat which occurs when an adversary decides to modify the circuit of the IP cores. These trojans, when triggered, release a payload that generally results in heat dissipation,excessive loss of battery, or overall decrease in performance which are hazardous for the end consumer.

A Trojan might be parametric or functional in nature. A Trojan is functional if the adversary modifies the original chip design by adding or removing transistors or gates. The parametric Trojan changes the original circuitry, such as weakening flip-flops, exposing the chip to radiation, thinning wires, or transistors, or employing Focused Ion-Beams (FIB) to impair chip dependability.
The possibility of a significant, malevolent design change is particularly pertinent to government entities. Resolving doubts regarding hardware integrity is one method of reducing technological vulnerabilities in an economy's military, banking, energy, and political sectors. Due to the increase of the supply chains to 3PIP increase of trojans have become common so new detection techniques should also be risen to identify when an adversary has malfunctioned our circuit.

Chapter 3

Proposed Work

3.1 Overall Workflow

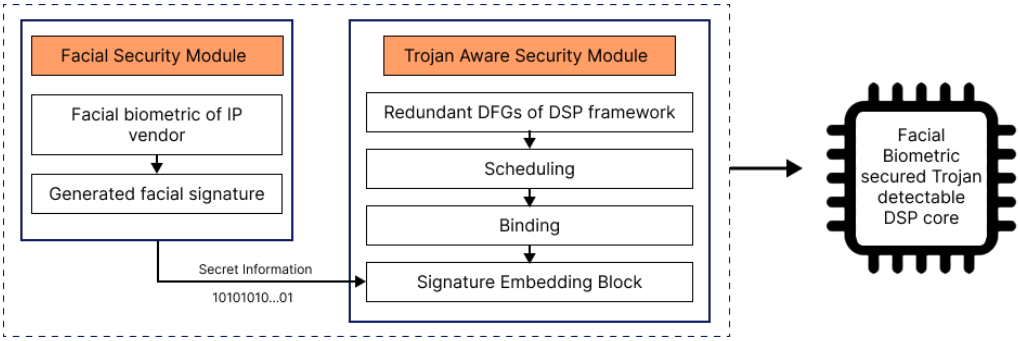

Figure 3.1: Proposed Methodology

3.2 Detailed Explanation

3.2.1 Input DFG

Let us consider a 4-point DCT.

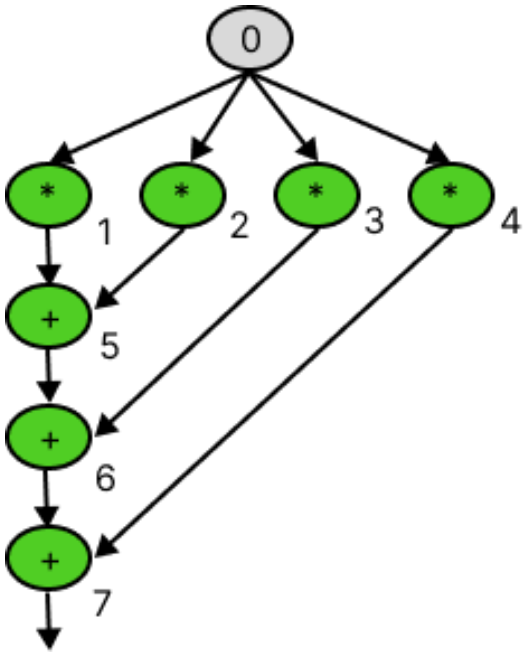

Figure 3.2:Data flow graph of a 4-point DCT

In equivalent textual representation. it will be as follows :-

*,0,0,1
*,0,0,2
*,0,0,3
*,0,0,4
+,1,2,5
+,5,3,6
+,6,4,7

3.2.2 Constructing a Trojan Secured Design

In order to construct a Trojan Secured DSP design, first we generate a data flow graph(DFG) of the corresponding DSP framework and we generate a sister unit of the same DFG by duplicating the operations of the original unit.

Together, these two components fulfil the function of the Trojan Secured Design.

3.2.3 Scheduling DFG of the Trojan Secured Design

In order to schedule the operations, we have used LIST scheduling with user defined resource constraints. As it was mentioned earlier, the DFG of the original DSP design will be duplicated and scheduled together.

If the dependency in the original unit is *,a,b,c in duplicate unit it will be *,a',b',c'

Let us consider 4-point DCT with resource constraint of 2 multipliers and 1 adder and with a redundant duplicate unit.

This will be the scheduled DFG as per LIST algorithm

Control Step 1 ***,1 *,2**
Control Step 2 **+,5 *,3 *,4**
Control Step 3 **+,6 *,1' *,2'**
Control Step 4 **+,7 *,3' *,4'**
Control Step 5 **+,5'**
Control Step 6 **+,6'**
Control Step 7 **+,7'**

After scheduling we allocate storage variables and construct a register allocation table :- (Vx represents a storagee variable and Ry represents a register)

Table 3.1: Register allocation table before embedding Facial Signature

	R1	R2	R3	R4	R5	R6	R7	R8
C0	V0	V1	–	–	–	–	–	–
C1	V2	V3	V8	V9	–	–	–	–
C2	V4	V5	V10	V11	V12	–	–	–
C3	V6	V7	V13	V11	V15	V16	–	–
C4	V14	V17	V18	–	V15	V16	–	–
C5	V19	V17	V18	–	–	–	–	–
C6	V20	–	V18	–	–	–	–	–
C7	V21	–	–	–	–	–	–	–

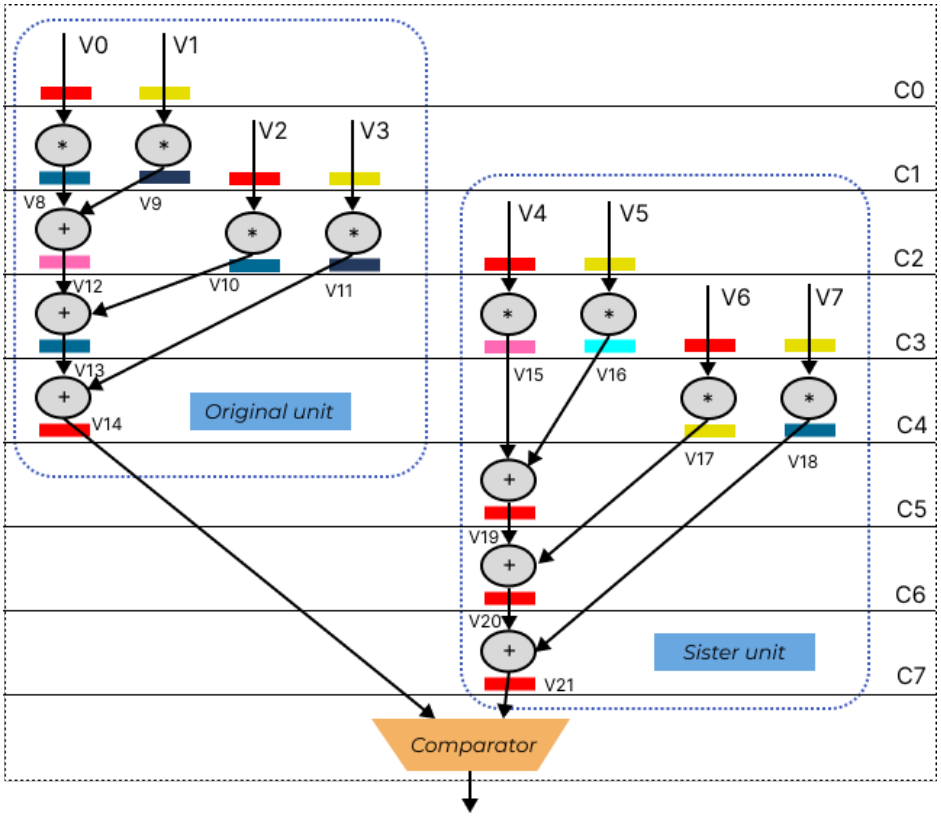

Figure 3.3: Scheduled DFG of Trojan secured DSP design of 4-point DCT

After register allocation, when resources are allocated. To distribute the resources to each unit, a distinct multivendor policy is used. As an illustration, if vendor X1 gives resources to the original unit, vendor X2 will do the same for the sister unit, and vice versa. If the Trojan is implanted in the design, this aids in its detection. Let's imagine that vendor X1 included the Trojan knowingly in the design (let say the multiplier operation performs the addition). The vendor X2 end, however, cannot exhibit the same behaviour.

When these two units (one with true functionality and another with malicious trojan) arrives at system on chip (SoC) integrator end, both units are combined to generate the final design. These two units will be supplied with the same input and comparator will pass the output only if they have same output, if they have different output, the comparator will report the presence of a trojan.

Furthermore, there are extremely few chances of a Trojan being undetected because both IP providers are unaware of one another and it is quite unlikely that the same type of Trojan could exist at the same design level (node) and produce results that are similar.

3.2.4 Generating Facial Bio-metric Signature

The facial security module of the design corresponds to generating a facial bio metric signature of the IP vendor to be embedded into the Trojan protected DSP design. The detailed process is as follows:-

Capturing the facial bio-metric Signature of IP vendor

The Facial Image of the corresponding IP designer is captured using a high definition Image capturing device. The captures image is places on a vendor specific grid size to generate bio-metric information accurately.

Generating Facial feature set and placing nodal points

Based on the IP vendor's chosen facial features, generation of nodal points will take place. Nodal points will be assigned a distinct naming convention.

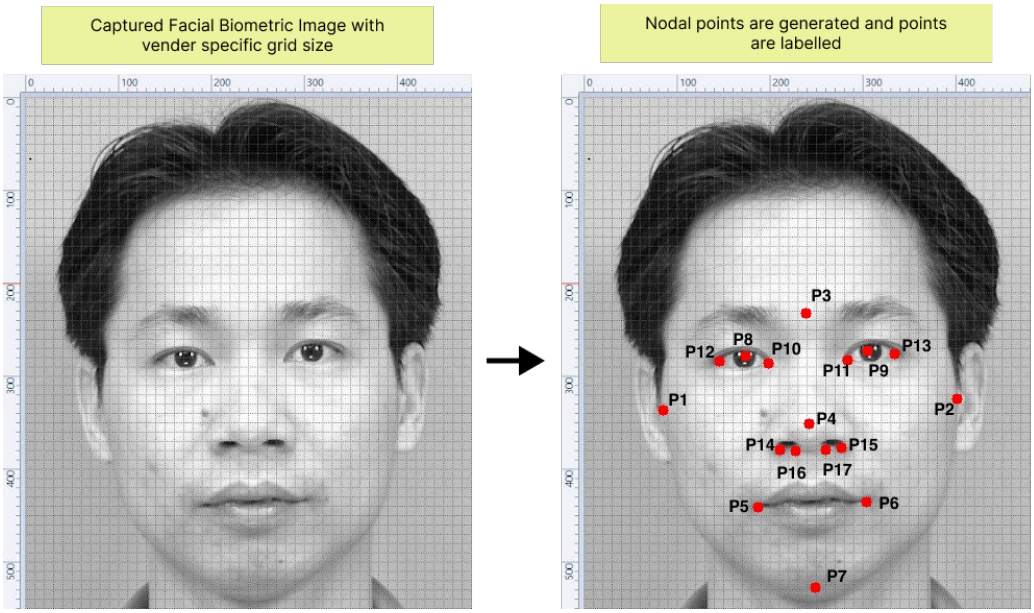

Figure 3.4: Generating Image with Nodal Points

Generating Facial Image with selected feature set

Facial image with IP vendor's selected feature set is generated as in fig(3.5). A facial feature here denoted as distance between 2 nodal points. For example - Nodal points P1 and P2 represent the Width of face(WF)

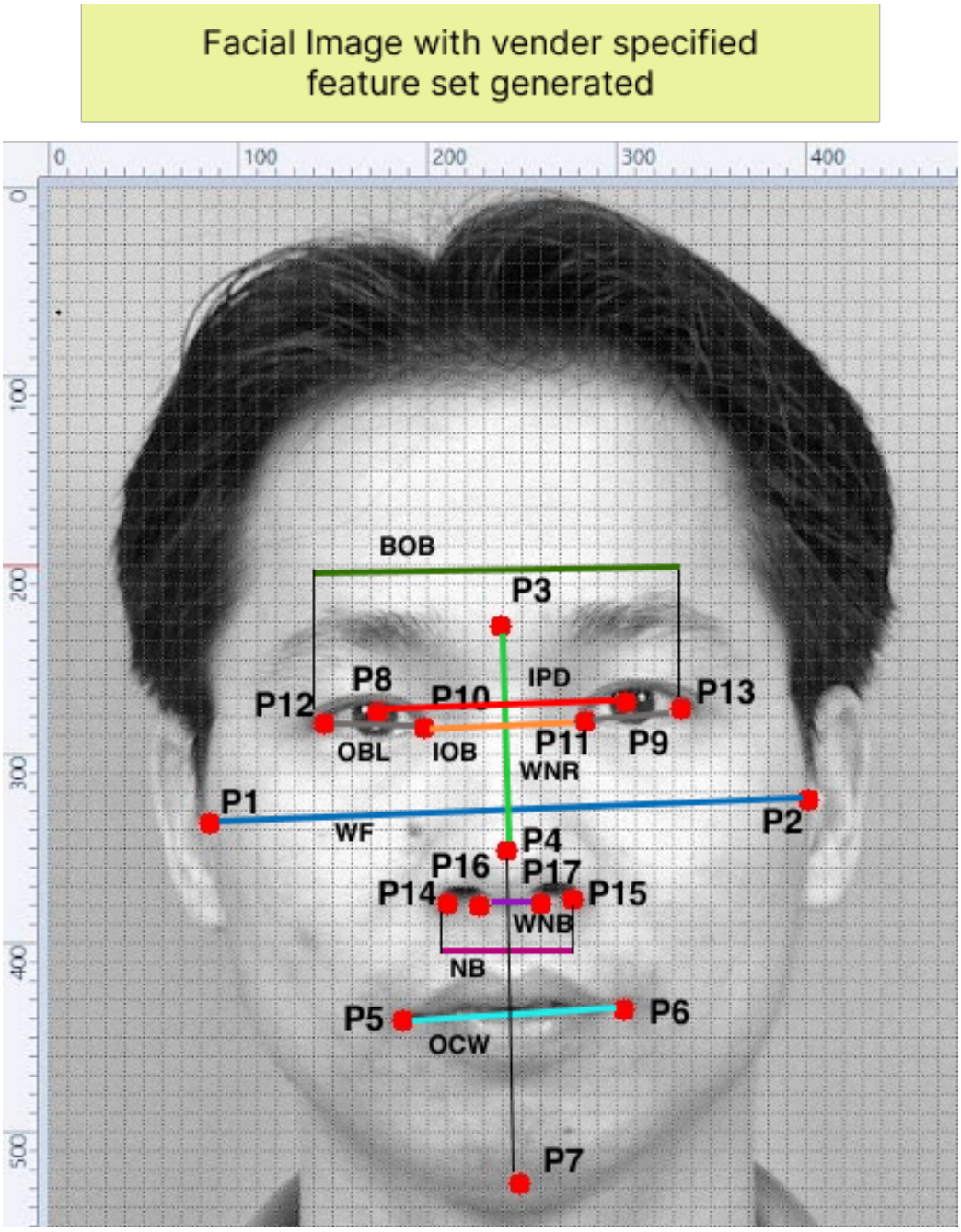

Figure 3.5: Facial Image with selected feature set

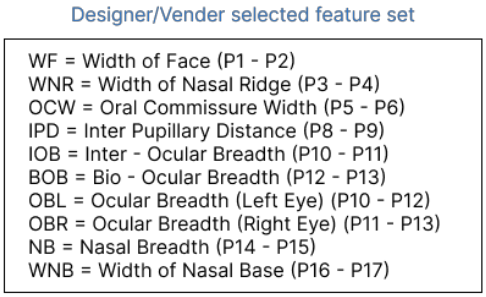

Figure 3.6: Feature set selected by vendor

Feature Dimensions for chosen features between nodal points

Facial features	Feature dimension (Manhattan Distance)= |x2-x1|+|y2-y1|	Binarized Features
WF	293	100100101
BOB	175	10101111
IOB	79	1001111
OBL	49	110001
OBR	51	110011
WNR	109	1101101
WNB	30	11110
NB	61	111101
OCW	109	1101101
IPD	122	1111010

Figure 3.7: Feature dimensions

Generating Facial Signature

To generate Facial Bio-metric Signature, as a digital biometric template, the dimensions are calculated. Between the nodal points corresponding to each face feature, we calculate the Manhattan distance.

Each feature dimension is then converted to its binarized state. The binarized data corresponding to each facial characteristic is then concatenated to create the facial signature.

However, distinct facial biometric signatures can be created based on the various concatenation orderings of facial elements. Additionally, the IP provider can choose the face signature of the appropriate strength (size) by choosing or excluding certain facial features from the final signature that will be incorporated into the design. As an illustration, the generated facial signature for the concatenation of the following face features will look like this:

(BOB) (IOB) (OBL) (OBR) (WNR) (WF) (WNB) (NB) (OCW) (IPD). Where (X1, X2), (Y1, Y2) stands for the coordinates corresponding to both nodal points of each feature, and "" denotes the concatenation operator. The final facial signature that results is:

10101111100111111000111001111011011001001011111011110111011011111010

3.2.5 Embedding Facial Signature into DSP design and RTL generation

The secret facial bio-metric data of the legitimate IP owner is inserted into target design by the signature embedding block. The resulting face signature is transformed into the appropriate hardware security constraints.This hardware security constraint creation is based on the DSP design framework (which specifies the number of node pairs) and encoding rule. For example, if the encoding rule supplied by the vendor is:

The embedding of security constraints between odd odd storage variable pairs is dictated by signature bit '1', while the embedding of security constraints between even-even storage variable pairs is dictated by signature bit '0'.

The resulting hardware security constraints derived for bit-'1'and bit-'0' are as follows:

<V1,V3>,<V1,V5>,<V1,V7>,<V1,V9>,<V1,V11>,<V1,V13>,<V1,V15>,<V1,V17>,<V1,V19>,<V1,V21>,<V3,V5>,<V3,V7>,<V3,V9>,<V3,V11>,<V3,V13>,<V3,V15>,<V3,V17>,<V3,V19>,<V3,V21>,<V5,V7>,<V5,V9>,<V5,V11>,<V5,V13>,<V5,V15>,<V5,V17>,<V5,V19>,<V5,V21>,<V7,V9>,<V7,V11>,<V7,V13>,<V7,V15>,<V7,V17>,<V7,V19>,<V7,V21>,<V9,V11>,<V9,V13>,<V9,V15>,<V9,V17>,<V9,V19>,<V9,V21>,<V11,V13>,<V11,V15>,<V11,V17>,<V11,V19>,<V11,V21>,<V13,V15>,<V13,V17>,<V13,V19>,<V13,V21>,<V15,V17>,<V15,V19>,<V15,V21>,<V17,V19>,<V17,V21>,<V19,V21>

Figure 3.8: Constraints dictated by '1' bit

<V0,V2>,<V0,V4>,<V0,V6>,<V0,V8>,<V0,V10>,<V0,V12>,<V0,V14>,<V0,V16>,<V0,V18>,<V0,V20>,<V2,V4>,<V2,V6>,<V2,V8>,<V2,V10>,<V2,V12>,<V2,V14>,<V2,V16>,<V2,V18>,<V2,V20>,<V4,V6>,<V4,V8>,<V4,V10>,<V4,V12>,<V4,V14>,<V4,V16>,<V4,V18>,<V4,V20>,<V6,V8>,<V6,V10>,<V6,V12>,<V6,V14>,<V6,V16>,<V6,V18>,<V6,V20>,<V8,V10>,<V8,V12>,<V8,V14>,<V8,V16>,<V8,V18>,<V8,V20>,<V10,V12>,<V10,V14>,<V10,V16>,<V10,V18>,<V10,V20>,<V12,V14>,<V12,V16>,<V12,V18>,<V12,V20>,<V14,V16>,<V14,V18>,<V14,V20>,<V16,V18>,<V16,V20>,<V18,V20>

Figure 3.9: Constraints dictated by '0' bit

The security constraints will be added as edges to the register allocation, because if two storage variables share an edge, that means they wont be using same register. If two variables are using same register and now an edge is added between them, they wont be using the same register post embedding. After adding all these edges to the Trojan protected DSP design, we will allocate registers again.

After performing the embedding of security constraints in the from of edges, the register allocation table is as follows:

Table 3.2: Register allocation table after embedding Facial Signature

	R1	R2	R3	R4	R5	R6	R7	R8
C0	V0	V1	–	–	–	–	–	–
C1	V3	V2	–	V8	V9	–	–	–
C2	–	–	V4	V5	V10	V11	V12	–
C3	–	–	V7	V6	V16	V11	V13	V15
C4	–	–	V14	V18	V16	–	V17	V15
C5	–	–	–	V18	–	–	V17	V19
C6	–	–	V20	V18	–	–	–	–
C7	–	–	–	–	–	–	V21	–

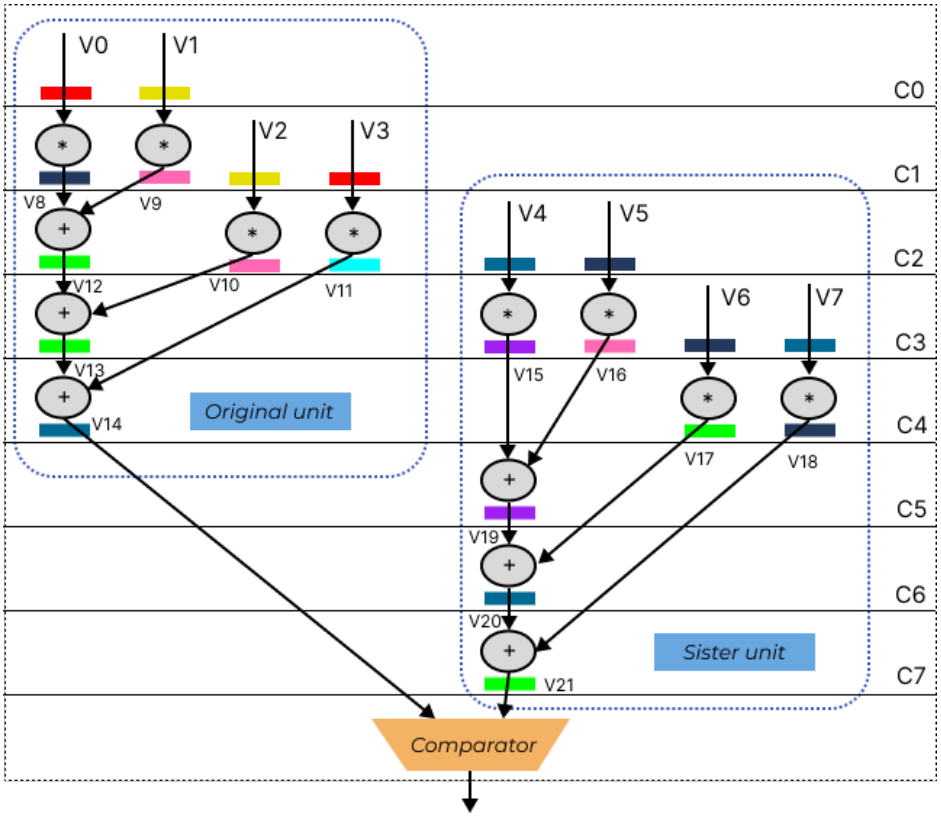

Figure 3.10: Scheduled DFG of Facial Signature embedded Trojan secured DSP design of 4-point DCT

3.3 Proposed Evaluation Models

3.3.1 Cost Analysis

The suggested methodology protects Trojan-secured design architecture by embedding security at a higher abstraction level of the design process, resulting in low design cost. Design cost (Φ) is formulated as follows [5]:

$$\phi = \eta_1 \left(\frac{\Omega_A}{\Omega_{Max}} \right) + \eta_2 \left(\frac{\mho_D}{\mho_{Max}} \right)$$

Where $'\Omega'_A$ and $'\mho'_D$ reflect the Trojan secured design's area and latency. $'\Omega_{Max}$ and $\mho_{Max}'$ denotes the maximum area and delay, whereas η_1 and η_2 imply the normalisation factors used to normalise the design area and latency.

Calculation - For our 4-point DCT example -
$\Omega_A = 346.0304$ $\Omega_{Max} = 648.0208$
$\mho_A = 1258.6132$ $\mho_{Max} = 2186.0124$

$\phi = 0.5548689833$

Table 3.3 summarises the design costs associated with various DSP systems. As shown in the table, there is no design overhead for enabling the protection of Trojan protected DSP designs against IP infringement using facial biometrics.

Table 3.3: Design costs of the approach

DSP benchmarks	Number of Registers	Design cost of trojan secured design	Design cost of generating trojan secured design protected against IP piracy using facial biometric	Overhead(%)
ARF	16	0.37	0.37	0%
JPEG	20	0.64	0.64	0%
FIR	16	0.39	0.39	0%
DCT	8	0.53	0.53	0%

3.3.2 Security Analysis

The suggested approach's security is assessed using the probability of coincidence (Pc) metric. The Pc value denotes the degree of ownership. If an adversary (belonging to) If a design firm (fab or fabless) falsely claims ownership rights, the actual IP owner must be able to prove ownership. The lower the Pc value, the greater the distinguishability between authentic and counterfeited designs in terms of hardware security limitations, signifying more unique and robust security. The Pc metric is defined as follows [5]:

$$Pc = \left(1 - \frac{1}{\kappa}\right)^{\chi}$$

Where "χ" and "κ" represent the strength (size) of security constraints implanted into the design during the higher abstraction level of the design phase, and the number of registers in the design, respectively. unprotected target design Table X shows the Pc comparison for a Trojan-secured 4-Point DCT design with varying facial signature strength. As shown in Table 3.4, increasing the strength of secret constraints decreases the Pc value. Fig. 3.11 also shows the Pc achieved with facial biometric security for various DSP benchmarks. As shown in Fig. 3.12, the proposed approach has a lower Pc than hardware steganography [6] and hardware watermarking [5].

Table 3.4: Pc comparison for trojan secured DSP 4-point DCT, corresponding to varying facial features

Facial Features	Constraints	Pc
5	40	4.7E-3
6	47	1.8E-3
8	62	2.5E-4
10	78	2.9E-5

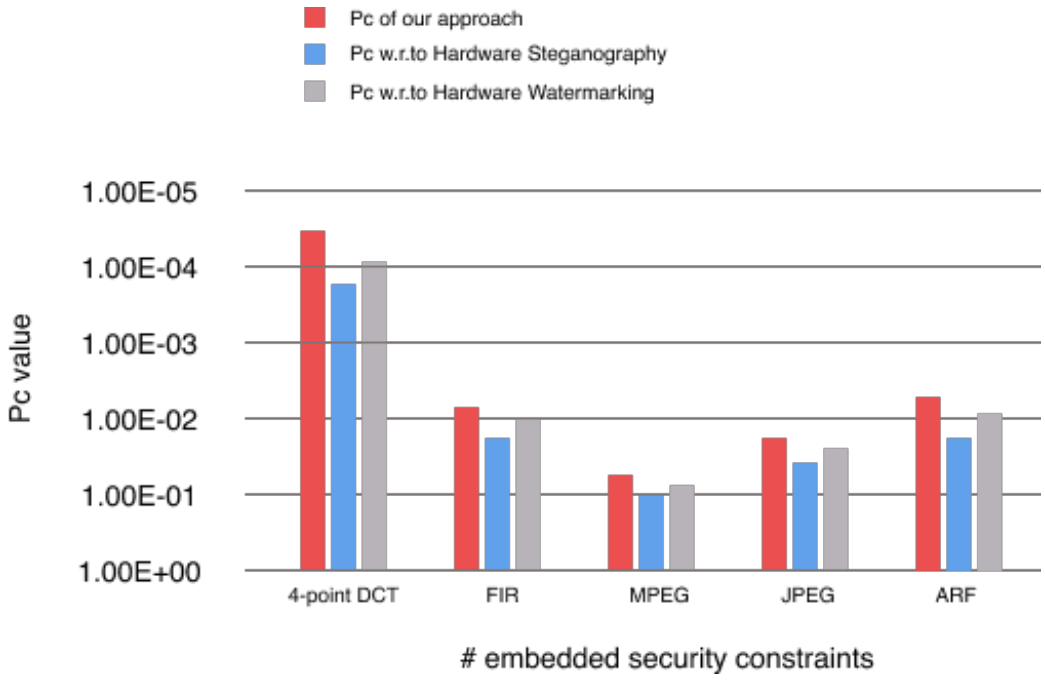

Figure 3.11: Pc comparison of the our approach corresponding to different DSP benchmarks

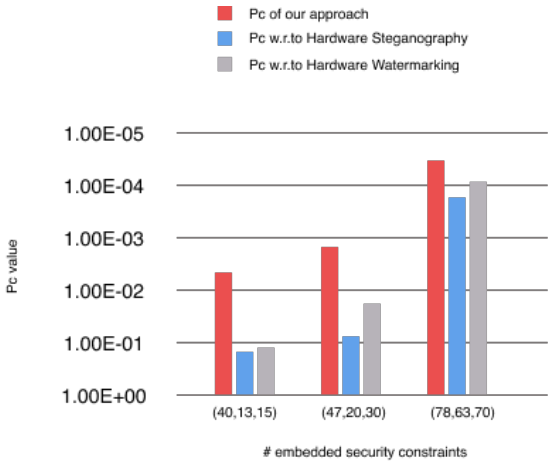

Figure 3.12: Pc comparison of Trojan Secured 4-point DCT core for varying security constraints

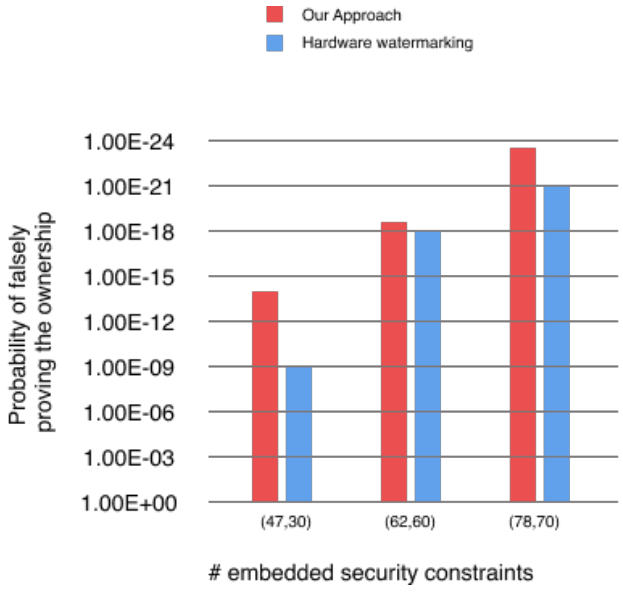

Figure 3.13: Probability for an attacker to search exact signature combination to prove ownership falsely

Chapter 4

Conclusion

Using facial biometrics, this work implemented an approach for allowing the robust security of Trojan-secured DSP designs against the dangers of IP counterfeiting and piracy.

The methodology put forward a way to detected designs with hardware trojans by implementing a multi vendor policy to allocate resources. This way the design is protected from Hardware Trojans with minimum design cost overhead.

The methodology displayed enhanced security in terms of detecting counterfeited designs and assigning IP ownership to the legitimate IP seller using facial biometrics based security exclusively while incurring no design cost overhead.

This methodology also has better security metrics such as probability of coincidence and probability that an attacker can search the exact signature to claim ownership, compared to previous approaches like hardware steganography and hardware watermarking, because our approach generates more security constraints, thus providing more security against IP counterfeiting/piracy.

As a result, the suggested methodology assures that only legitimate and Trojan-free designs are included into consumer electronics systems.

4.1 Future Scope

The methodology can be extended to other multi-modal biometric approaches for protecting Trojan secured DSP designs such as approaches like retina based biometric approach, iris based biometric approach and fingerprint based biometric approach. This may enable more robust security to Trojan protected DSP designs due to the the potential of these approaches to produce more number of security constraints and also making them very hard to access,thus providing more secuirity to IP cores

Thus, multi-modal biometric based hardware security approaches provide a strong case for further exploration.

List of Figures

List of Tables

Chapter 5

References

1. M. T. Arafin, A. Stanley and P. Sharma, “Hardware-based anti- counterfeiting techniques for safeguarding supply chain integrity,” 2017 IEEE International Symposium on Circuits and Systems (ISCAS), 2017, pp. 1-4.

2. B. K. Mohanty and P. K. Meher, “A High-Performance FIR Filter Architecture for Fixed and Reconfigurable Applications,” IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 24, no. 2, pp. 444-452, Feb. 2016.

3. Bao, D. Forte and A. Srivastava, ”On Reverse Engineering-Based Hardware Trojan Detection,” in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 35, no. 1, pp. 49-57, Jan. 2016.

4. A. Sengupta, S. Bhadauria and S. P. Mohanty, ”TL-HLS: Methodology for Low Cost Hardware Trojan Security Aware Scheduling With Optimal Loop Unrolling Factor During High Level Synthesis,” in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 36, no. 4, pp. 655-668, April 2017.

5. A. Sengupta and S. Bhadauria, ”Exploring low cost optimal watermark for reusable IP cores during high level synthesis,” IEEE Access, vol. 4, pp. 2198–2215, 2016.

6. M. Rathor and A. Sengupta, “IP Core Steganography Using Switch Based Key-Driven Hash-Chaining and Encoding for Securing DSP Kernels Used in CE Systems,” IEEE Trans. Consum. Electron., vol. 66, no. 3, pp. 251-260, Aug. 2020.

7. A. Sengupta, E. R. Kumar, and N. P. Chandra, "Embedding digital signature using encrypted-hashing for protection of DSP cores in CE," IEEE Trans. Consum. Electron., vol. 65, no. 3, pp. 398–407, Aug. 2019.

8. A. Sengupta and M. Rathor, "Securing hardware accelerators for CE systems using biometric fingerprinting," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 28, no. 9, pp. 1979–1992, Sep. 2020.

9. Detecting Hardware Trojans with GateLevel InformationFlow Tracking, Wei Hu et al, IEEE publication, 2015