Secure FFT IP using C way Partitioning based Obfuscation and Fingerprint

Published in IEEE Design & Test

Anirban Sengupta and Rahul Chaurasia, "Secure FFT IP Using C-Way Partitioning-Based Obfuscation and Fingerprint," IEEE Design & Test, vol. 41, no. 5, pp. 55-64, Oct. 2024

INTRODUCTION

- The FFT is a fundamental building block used in signal processing systems, with applications ranging from Digital MODEMs, to ultrasound and Computed Tomography (CT) image reconstruction.

- However, no work focused on designing its secure IP design using HLS..

RELATED WORK

Sr. No.	Existing Work	Technique Used	Remarks
1.	A. Sengupta et. al., [6] (2019)	Steganography offers security through embedded stego constraints.	However, none of these approaches [6], [7] are robust as they can be compromised by an adversary by decoding auxiliary security parameters. Moreover, these approaches are not capable of hindering an attacker to perform RTL alteration, as they do not have provision for HLS-based RTL structural obfuscation, unlike proposed approach
2.	F. Koushanfar et. al., [7] (2005)	Multi-variable signature combinations for security	

A. Threat Model and Motivation of Proposed Methodology

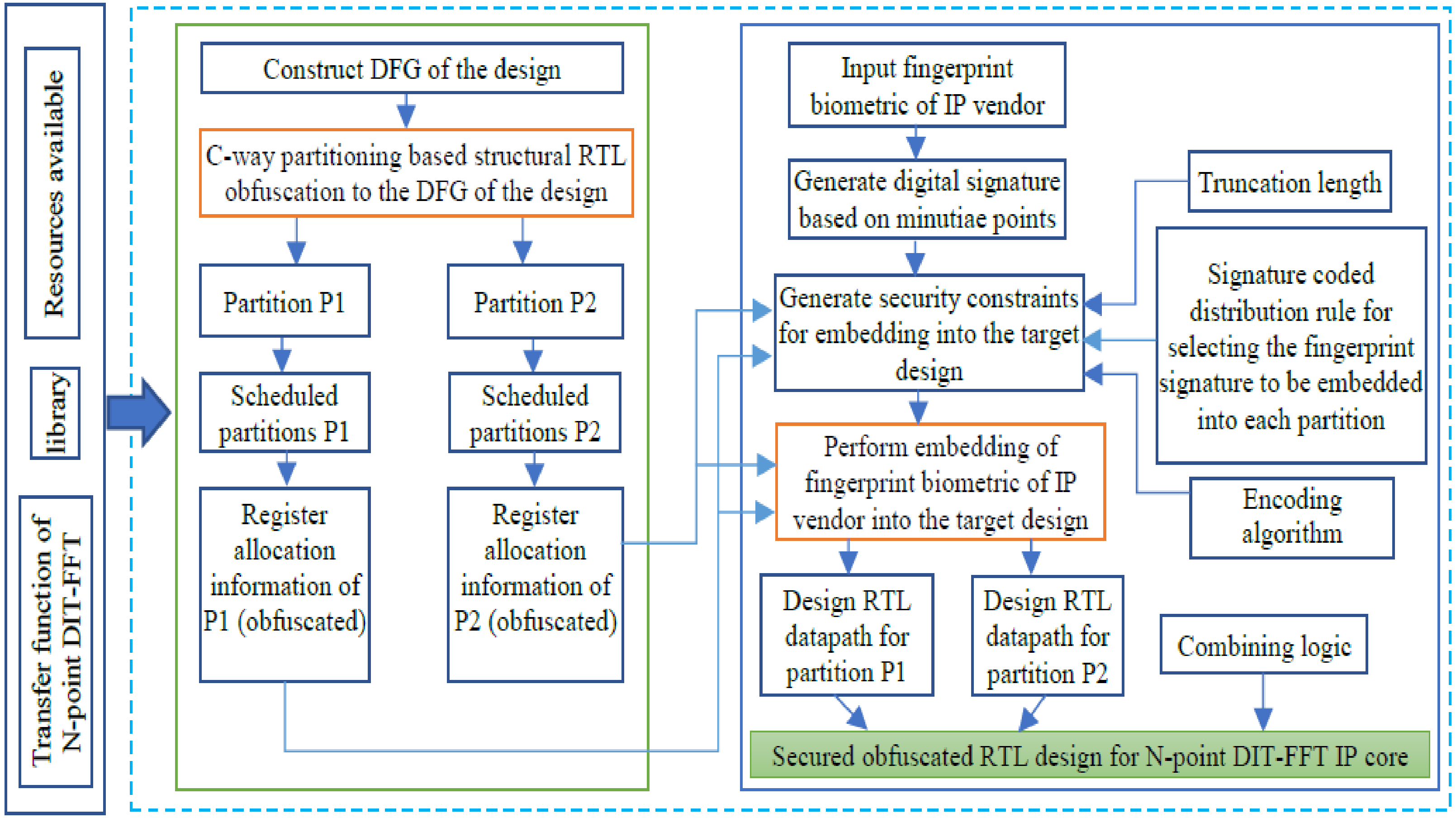
- In this paper, this threat is handled by employing C-way partitioning-based structural RTL obfuscation by an IP vendor in order to hide the functionality and structural interconnection of IP design.

- The effects of RTL obfuscation on layout obfuscation have been established in several literatures [2], [13]. Further as established in [6], [7], [8], an adversary in the SoC house, may attempt to perform IP piracy and falsely claim of IP.

- Therefore, these threats are tackled using proposed IP vendor fingerprint biometric as detective countermeasure.

PROPOSED WORK

A. Overview



Resources available
library
Transfer function of N-point DIT-FFT
Construct DFG of the design
C-way partitioning based structural RTL obfuscation to the DFG of the design
Partition P1
Partition P2
Scheduled partitions P1
Scheduled partitions P2
Register allocation information of P1 (obfuscated)
Register allocation information of P2 (obfuscated)
Input fingerprint biometric of IP vendor
Generate digital signature based on minutiae points
Truncation length
Generate security constraints for embedding into the target design
Signature coded distribution rule for selecting the fingerprint signature to be embedded into each partition
Perform embedding of fingerprint biometric of IP vendor into the target design
Encoding algorithm
Design RTL datapath for partition P1
Design RTL datapath for partition P2
Combining logic
Secured obfuscated RTL design for N-point DIT-FFT IP core

Fig. 1. Design flow of the proposed security methodology

PROPOSED WORK

B. Secured Scheduled FFT Design using Proposed Approach

- Transfer function of Discrete Fourier Transform (DFT) is represented as [8]:

$$X(k) = \sum x(n)\, W_N^{nk}, 0 \leq K \leq N - 1 \qquad (1)$$

$$X(k) = G(k) + W_N^k H(k), k = 0,1, \ldots\ldots \frac{N}{2} - 1 \quad (2)$$

$$PC_S = \left(1 - \frac{1}{R_B}\right)^x \qquad (6)$$

PROPOSED WORK

B. Secured Scheduled FFT Design using Proposed Approach

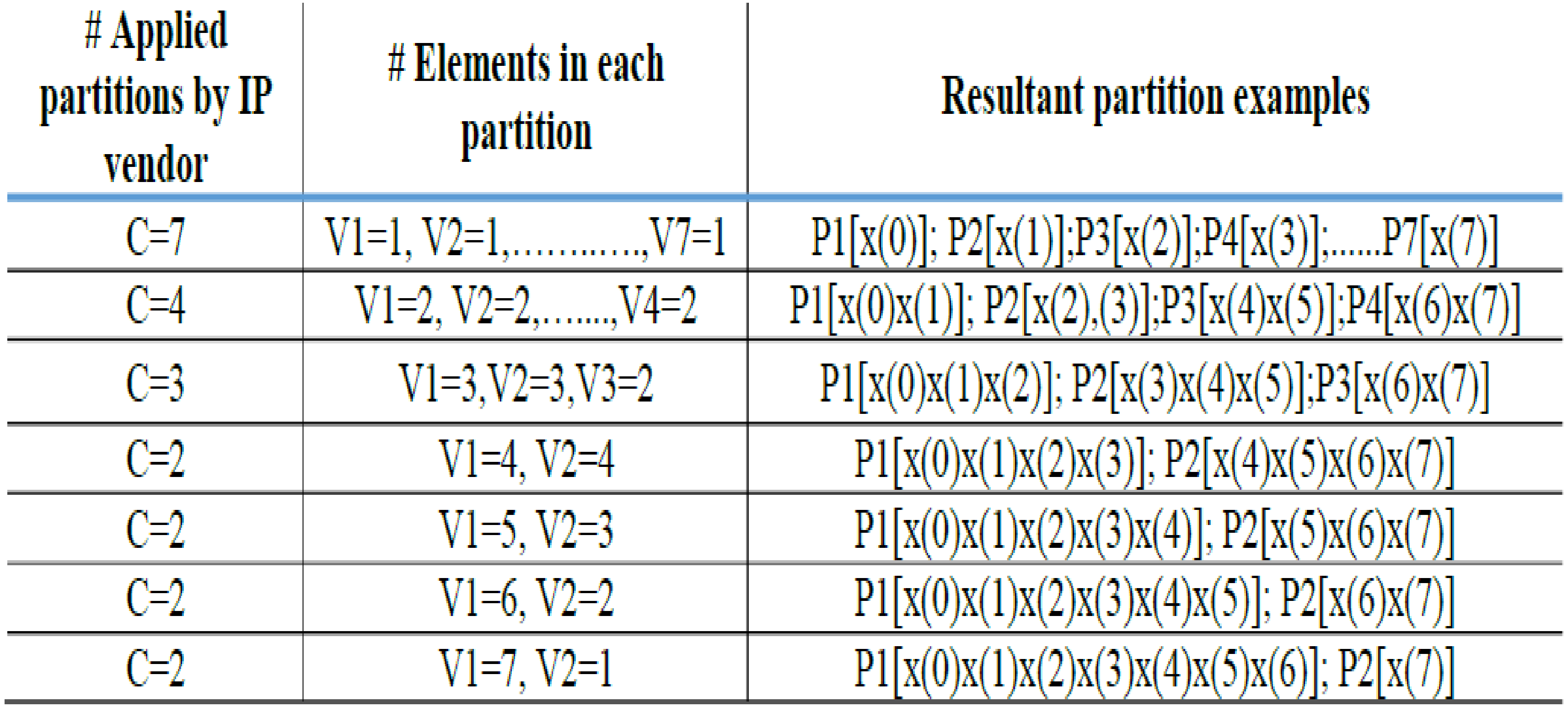
Now based on (2), the derived signal flow graph for N=8 has the following representations:

$$
\begin{gathered}
X(0)=G(0)+W80\ast H(0)\\
X(1)=G(1)+W81\ast H(1)\\
X(2)=G(2)+W82\ast H(2)\\
X(3)=G(3)+W83\ast H(3)\\
X(4)=G(0)+W84\ast H(0)\\
X(5)=G(1)+W85\ast H(1)\\
X(6)=G(2)+W86\ast H(2)\\
X(7)=G(3)+W87\ast H(3)
\end{gathered}
$$

PROPOSED WORK

Converting the transfer function into C-way partitioned structurally obfuscated scheduled design—

- The above equations are converted into CDFG, shown in Fig. 2 (a). An IP vendor can perform Q=2^(log2N)-1 types of partitioning depending on the number of data elements in each partition.

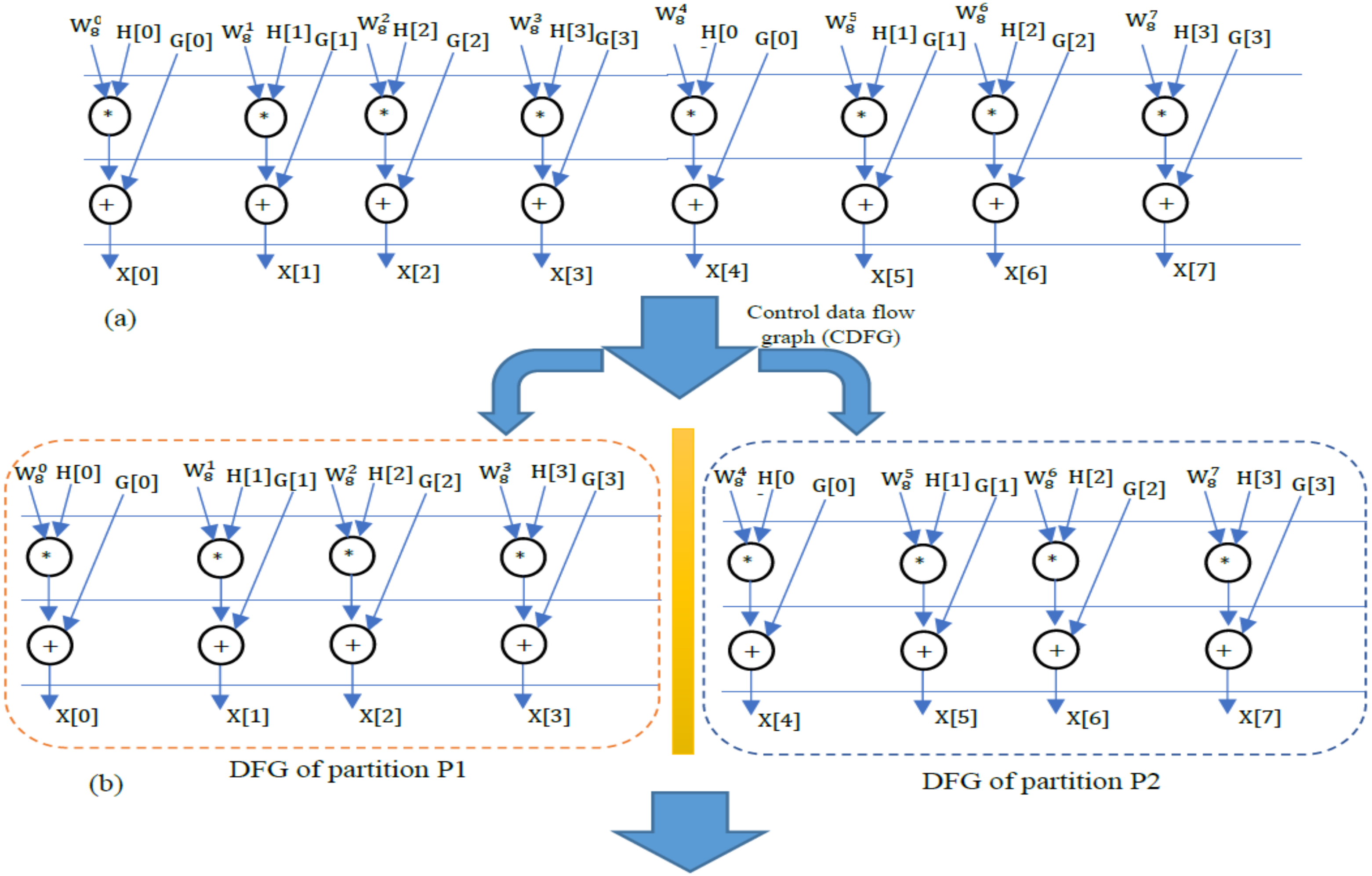


# Applied partitions by IP vendor	# Elements in each partition	Resultant partition examples
C=7	V1=1, V2=1,…………,V7=1	P1[x(0)]; P2[x(1)];P3[x(2)];P4[x(3)];......P7[x(7)]
C=4	V1=2, V2=2,……,V4=2	P1[x(0)x(1)]; P2[x(2),(3)];P3[x(4)x(5)];P4[x(6)x(7)]
C=3	V1=3,V2=3,V3=2	P1[x(0)x(1)x(2)]; P2[x(3)x(4)x(5)];P3[x(6)x(7)]
C=2	V1=4, V2=4	P1[x(0)x(1)x(2)x(3)]; P2[x(4)x(5)x(6)x(7)]
C=2	V1=5, V2=3	P1[x(0)x(1)x(2)x(3)x(4)]; P2[x(5)x(6)x(7)]
C=2	V1=6, V2=2	P1[x(0)x(1)x(2)x(3)x(4)x(5)]; P2[x(6)x(7)]
C=2	V1=7, V2=1	P1[x(0)x(1)x(2)x(3)x(4)x(5)x(6)]; P2[x(7)]

Table 1. Example of resulting partitions for Q=7

PROPOSED WORK

- **Converting the transfer function into C-way partitioned structurally obfuscated scheduled design—**



W_8^0 H[0] G[0] W_8^1 H[1] G[1] W_8^2 H[2] G[2] W_8^3 H[3] G[3] W_8^4 H[0 G[0] W_8^5 H[1] G[1] W_8^6 H[2] G[2] W_8^7 H[3] G[3]
*
+
X[0] X[1] X[2] X[3] X[4] X[5] X[6] X[7]

(a)

Control data flow graph (CDFG)

(b)

DFG of partition P1

DFG of partition P2

Fig. 2. (a) Data flow graph corresponding to DIT-FFT IP core for N=8

Fig. 2. (b) C- way partitioned DFG of FFT for C=2, V1=V2=4 and N=8

PROPOSED WORK

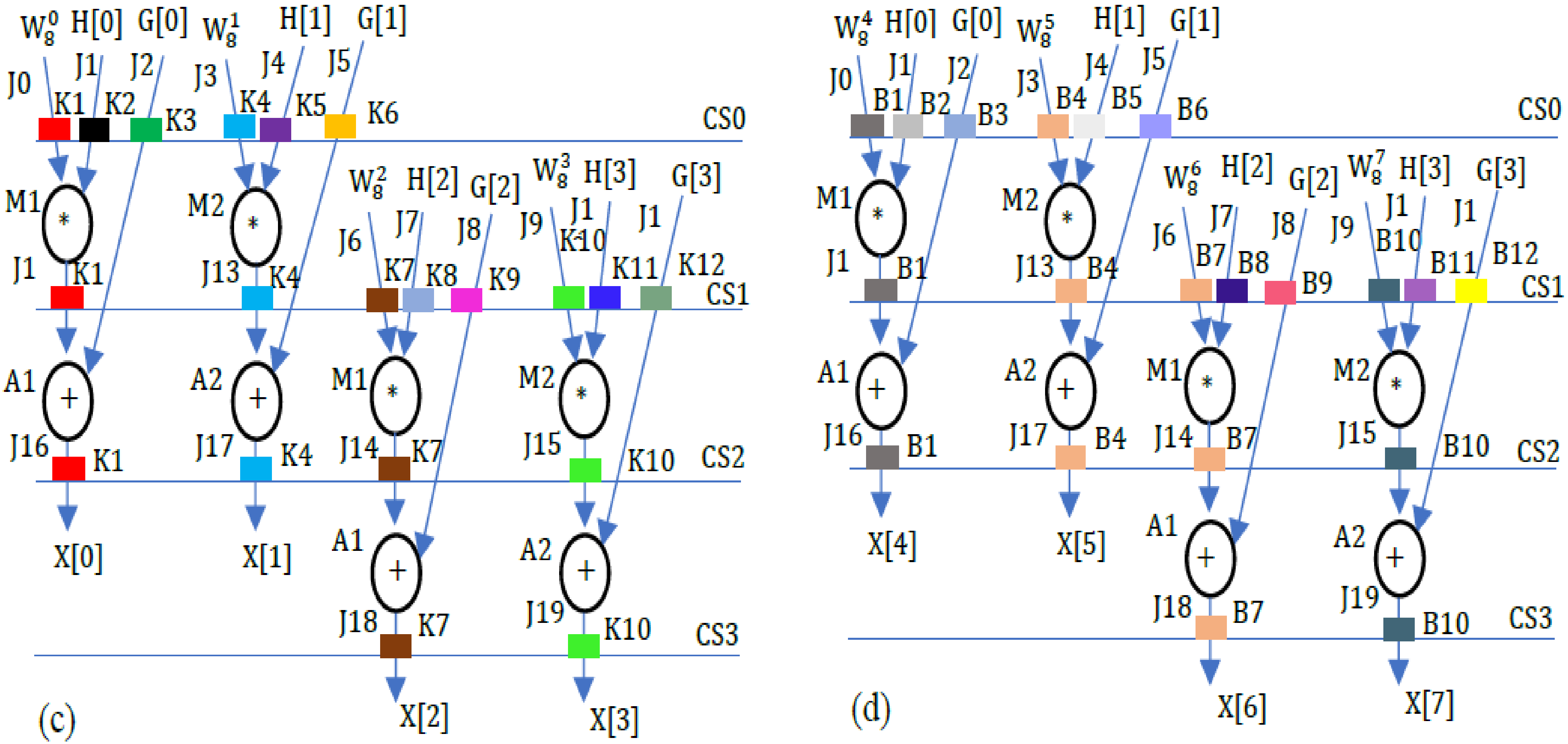


Fig. 2. (c) Scheduled DFG of structurally obfuscated partition P1 for 2A, 2M indicating storage variables (J0 to J19) and register (K1 to K12) allocation. **(d)** Scheduled DFG of structurally obfuscated partition P2 for 2A,2M indicating storage variables and register (B1 to B12) allocation.

PROPOSED WORK

C. Generating the Fingerprint Biometric Digital Template

The following subprocesses are executed (Fig. 3):

Sub-process-1**:** capture the fingerprint biometric of IP vendor using a scanning device.

Subprocess-2**:** preprocess the obtained image and perform the image enhancement using FFT in the obtained image to reconnect the broken ridges. Subsequently, perform the binarization and thinning process for smooth and accurate extraction of fingerprint minutiae points.

Subprocess-3**:** operate the thinned fingerprint image to extract the minutiae points features.

Subprocess-4**:** transform each of the minutiae points into their binarized form.

PROPOSED WORK

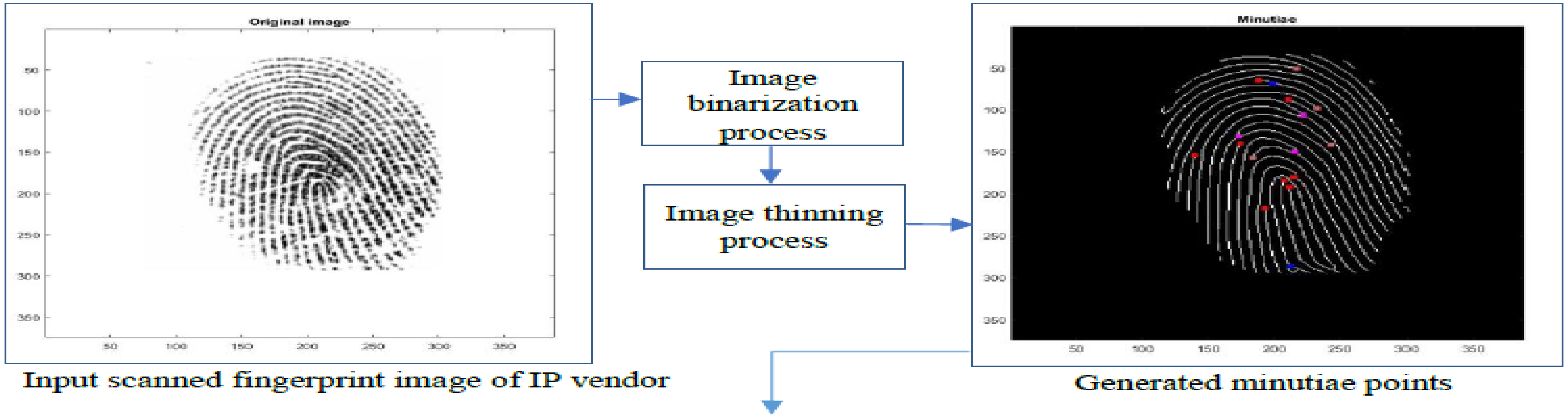


Image binarization process

Image thinning process

Input scanned fingerprint image of IP vendor

Generated minutiae points

# Minutiae points	x	y	m	Ridge angle (d) in degrees
1	217	51	1	23
3	199	69	3	193
4	211	88	1	210
5	233	98	1	45
6	222	106	3	39
7	173	131	3	130
8	175	140	1	295
9	243	142	1	59
10	216	150	3	45
11	140	154	1	281
12	184	157	1	111
13	215	180	1	225
14	208	184	1	262
15	212	192	1	260
16	193	217	1	267
17	213	286	3	235

Binarize representation of minutiae points

11011001110011110111.......110101011 0001 11101111101011 fingerprint template)

CODED DISTRIBUTION RULES

Signature strength (bits)	Rule	Encoded meaning
50	$(F_{25} ⧺ L_{25})$	Embed first 25 bits into P1 and last 25 bits into P2
-	-	--
-	$(F_z ⧺ L_{50-z})$	Embed first Z bits into P1 and last (50-Z) bits into P2
-	-	
400	$(F_{200} ⧺ L_{200})$	Embed first 200 bits into P1 and last 200 bits into P2
-	$(F_z ⧺ L_{400-z})$	--

Fig. 3. Generation of coded fingerprint signature

PROPOSED WORK

D. Generating the Hardware Security Constraints from Obfuscated Scheduled Design and Fingerprint Biometric .

For the sake of brevity, we are considering the fingerprint signature strength of 100 bits (including first 50 and last 50 bits of fingerprint signature). The first 50 bits of the generated fingerprint signature are:

“11011001-110011-1-10111-10111100-1000001-1-101100101-11000”.

D. Generating the Hardware Security Constraints from Obfuscated Scheduled Design and Fingerprint Biometric

IP Vendor's Encoding rule:

Rule-1: generate a security constraint by forming the storage variable pair (even-even) corresponding to signature bit '0'.

Rule-2: generate a security constraint by forming the storage variable pair (odd-odd) corresponding to signature bit '1'.

PROPOSED WORK

E. Generating the Secured RTL Datapath for N-point DIT-FFT Design with Embedded Fingerprint Biometric

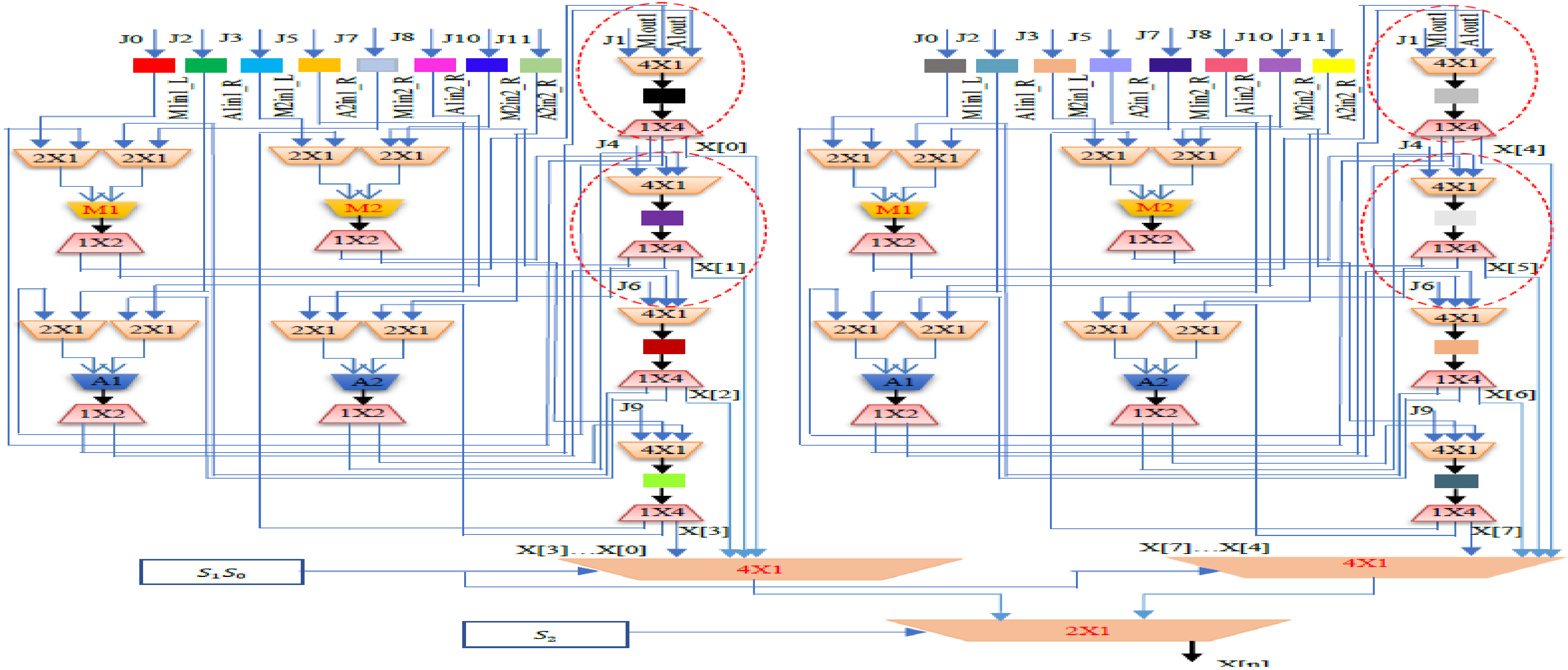
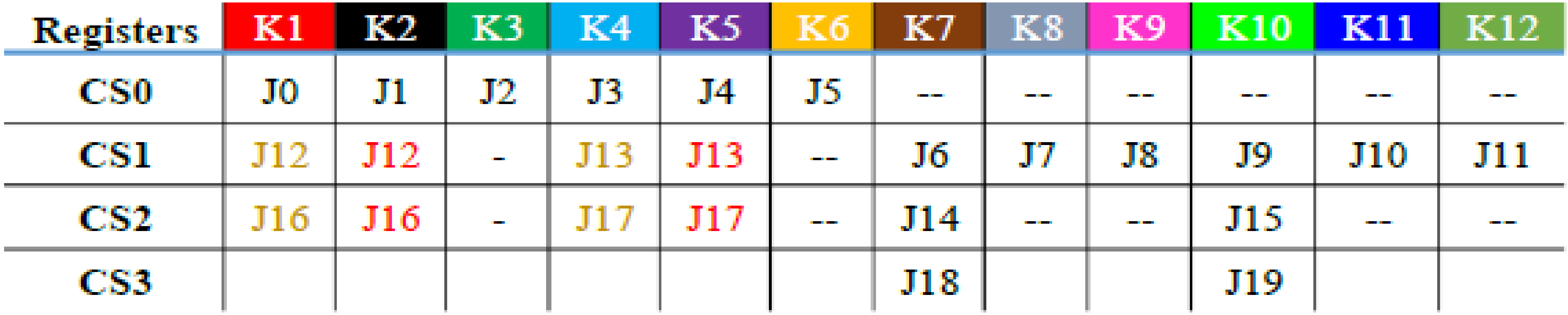


Fig. 4. Secured RTL Datapath design of N-point DIT-FFT (for N=8)

PROPOSED WORK

E. Generating the Secured RTL Datapath for N-point DIT-FFT Design with Embedded Fingerprint Biometric

Table 1. Locally altered register allocation information (post embedding first fifty bits of fingerprint signature into partition-P1).



Registers	K1	K2	K3	K4	K5	K6	K7	K8	K9	K10	K11	K12
CS0	J0	J1	J2	J3	J4	J5	--	--	--	--	--	--
CS1	J12	J12	-	J13	J13	--	J6	J7	J8	J9	J10	J11
CS2	J16	J16	-	J17	J17	--	J14	--	--	J15	--	--
CS3							J18			J19		

IMPLEMENTATION RESULTS

A. Analysis of Security due to Structural Obfuscation

- The hardness of structural obfuscation has been evaluated in [1], [5] and [11], using standard 'power of obfuscation' metric. 'Power of obfuscation (Pobf)' is expressed using following equations indicating normalized value between 0 to 1:'

$$\boldsymbol{P^{obf} = \frac{n}{n^T}} \qquad (3)$$

- Lower the probability, harder it is for an adversary to combine the partitions to the retrieve IP RTL design, during RE attempt. It can be represented as:

$$\boldsymbol{P_T = \frac{1}{2^{(log_2 N)}-1} \times \frac{1}{C}} \qquad (4)$$

Table II.A reports the P^{obf} and P_T of the proposed approach

IMPLEMENTATION RESULTS

A. Analysis of Security due to Structural Obfuscation

Table II. The P^{obf} and P_T of the proposed approach

N=8 (8-point FFT)			N=16 (16-point FFT)			N=32 (32-point FFT)		
C	P^{obf}	P_T	C	P^{obf}	P_T	C	P^{obf}	P_T
2	0.125	0.07	2	0.062	0.03	4	0.062	0.008
3	0.187	0.04	4	0.125	0.01	8	0.125	0.004
4	0.25	0.03	8	0.25	0.008	15	0.234	0.002
7	0.437	0.02	12	0.375	0.005	24	0.375	0.001
8	0.5	0.01	16	0.5	0.004	48	0.75	0.0006

IMPLEMENTATION RESULTS

A. Analysis of Security due to Structural Obfuscation

- Further, the attacker's effort needed in terms of de-obfuscating the gate level IP design is measured in terms of probability of identifying all obfuscated gates using brute force.

- The probability of identifying all obfuscated gates, due to C-way partitioning, from an attacker's perspective during applying brute force attack is given as:

$$P_G^T = \pi_{i=1}^{S^C}\left(\frac{1}{|G_S^T|}\right) \qquad (5)$$

- The affected gates (SC) due to proposed C-way partitioning resulting into structural obfuscation and P_G^T is presented in Table II.B.

IMPLEMENTATION RESULTS

A. Analysis of Security due to Structural Obfuscation

Table II.B. The affected gates (SC) due to proposed C-way partitioning resulting into structural obfuscation and P_G^T

Finger print Signature Size	*Un − obfuscated design gate count* (G_U^T)	Proposed Secured design gate count	Affected gates due to proposed C-way partitioning $S^C =$ $\lvert G_S^T - G_U^T \rvert$	Attacker's effort in de-obfuscating the proposed IP design (P_g^T)
		Post obfuscation without || with Fingerprint (G_S^T)		
50	4288	9502 || 9502	5214	$1 \div (9502)^{5214}$
100	4288	9502 || 9502	5214	$1 \div (9502)^{5214}$
150	4288	9502 || 9502	5214	$1 \div (9502)^{5214}$
200	4288	9502 || 9502	5214	$1 \div (9502)^{5214}$
400	4288	9502 || 9502	5214	$1 \div (9502)^{5214}$

IMPLEMENTATION RESULTS

B. Analysis of Security due to Fingerprint

- Probability of Coincidence **:**

$$\boldsymbol{PCs} = \left(\mathbf{1} - \frac{\mathbf{1}}{\boldsymbol{R_B}}\right)^{\boldsymbol{X}} \qquad (6)$$

Where '$\boldsymbol{R_B}$' and '$\boldsymbol{x}$' designate the count of storage elements in baseline design and number of generated security constraints.

- Tamper tolerance :

$$\boldsymbol{TT_A} = \boldsymbol{V^X} \qquad (7)$$

*Where, '**V**' represents the number of distinct bits of the signature.*

IMPLEMENTATION RESULTS

C. Entropy Estimation of the Proposed Approach

- The entropy of the proposed approach can be determined as follows:

$$E_t = \frac{1}{n!} \times \frac{1}{m!} \times \frac{1}{2^x} \qquad (8)$$

Where 'n' indicates the number of features of each minutia,
'm' indicates total number of minutiae points of fingerprint,
'x' indicates strength of signature and
'2^X' *indicates total signature space.*

IMPLEMENTATION RESULTS

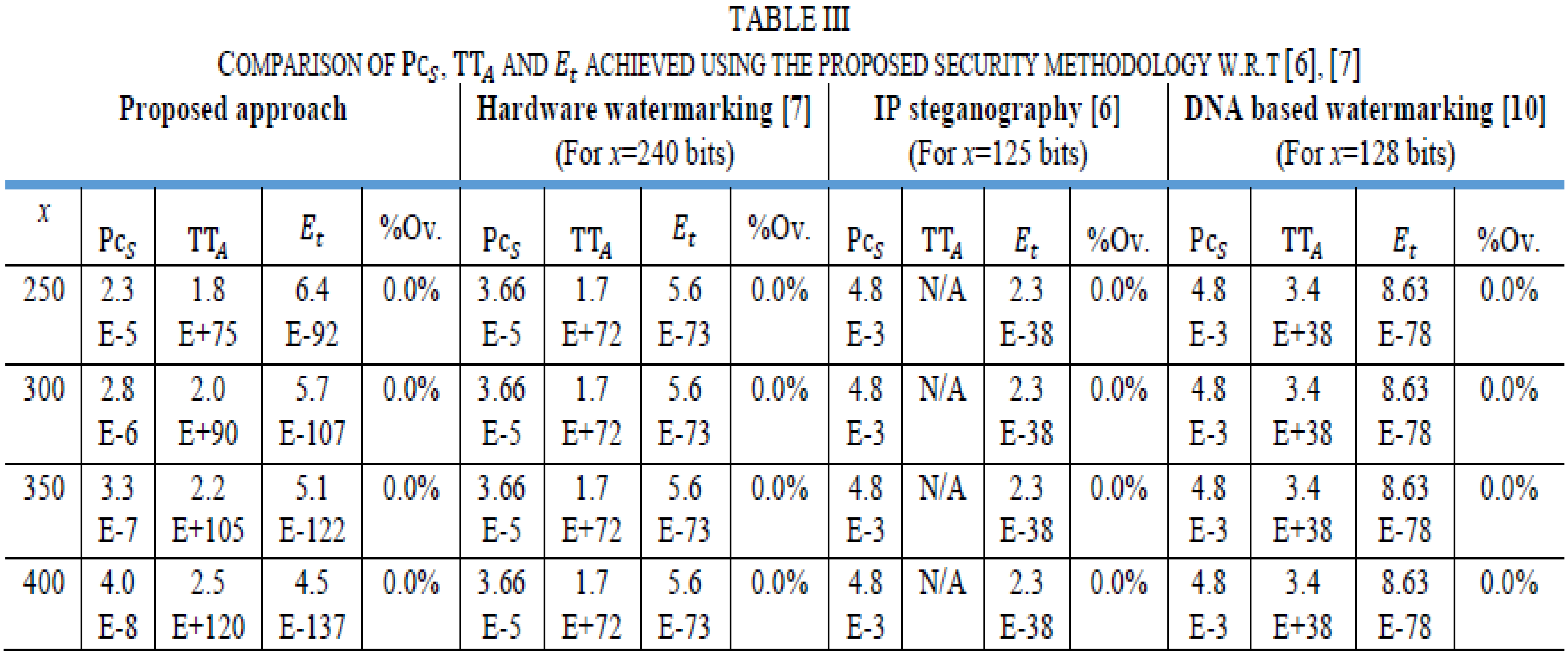


TABLE III

COMPARISON OF Pc_S, TT_A AND E_t ACHIEVED USING THE PROPOSED SECURITY METHODOLOGY W.R.T [6], [7]

	Proposed approach				Hardware watermarking [7] (For x=240 bits)				IP steganography [6] (For x=125 bits)				DNA based watermarking [10] (For x=128 bits)			
x	Pc_S	TT_A	E_t	%Ov.	Pc_S	TT_A	E_t	%Ov.	Pc_S	TT_A	E_t	%Ov.	Pc_S	TT_A	E_t	%Ov.
250	2.3 E-5	1.8 E+75	6.4 E-92	0.0%	3.66 E-5	1.7 E+72	5.6 E-73	0.0%	4.8 E-3	N/A	2.3 E-38	0.0%	4.8 E-3	3.4 E+38	8.63 E-78	0.0%
300	2.8 E-6	2.0 E+90	5.7 E-107	0.0%	3.66 E-5	1.7 E+72	5.6 E-73	0.0%	4.8 E-3	N/A	2.3 E-38	0.0%	4.8 E-3	3.4 E+38	8.63 E-78	0.0%
350	3.3 E-7	2.2 E+105	5.1 E-122	0.0%	3.66 E-5	1.7 E+72	5.6 E-73	0.0%	4.8 E-3	N/A	2.3 E-38	0.0%	4.8 E-3	3.4 E+38	8.63 E-78	0.0%
400	4.0 E-8	2.5 E+120	4.5 E-137	0.0%	3.66 E-5	1.7 E+72	5.6 E-73	0.0%	4.8 E-3	N/A	2.3 E-38	0.0%	4.8 E-3	3.4 E+38	8.63 E-78	0.0%

IMPLEMENTATION RESULTS

D. Detection of IP Piracy

- In the proposed approach, secret security constraints of biometric signature are extracted from the suspected chip design and matched with the originally embedded biometric security constraints of the IP design.
- From the extracted layout design file of the suspected chip, through backward engineering, the IP core RTL is obtained.
- Finally, the secret security constraints of biometric signature are extracted from the IP RTL file for matching [7], [10].
- In case of complete matching, IP piracy is detected.

REFERENCE

[1] A. Sengupta, S. Neema, S. Harsha, S. P Mohanty, M. K. Naskar, “Obfuscation of Fault Secured DSP Design through Hybrid Transformation”, Proceedings of 17th IEEE Computer Society Annual Symposium on VLSI (ISVLSI), Hong Kong, 2018, pp. 732 - 737.

[2] A. Vijayakumar, V. C. Patil, D. E. Holcomb, C. Paar and S. Kundu, “Physical Design Obfuscation of Hardware: A Comprehensive Investigation of Device and Logic-Level Techniques,” IEEE Trans. Inf. Forensics Secur., vol. 12, no. 1, pp. 64-77, Jan. 2017.

[3] Lao, Y. & Parhi, K., “Protecting DSP circuits through obfuscation,” Proceedings - IEEE International Symposium on Circuits and Systems (ISCAS), 2014, 798-801.

[4] J.Rajendran, A. Ali, O. Sinanoglu, R. Karri, “Belling the CAD: Toward Security-Centric Electronic System Design”, IEEE Trans. Comput-Aided Design Integr. Circuits Syst., Volume 34, Issue 11, 2015, pp 1756–1769.

[5] A. Sengupta, D. Roy, S. P. Mohanty and P. Corcoran, “DSP design protection in CE through algorithmic transformation based structural obfuscation,” IEEE Trans. Consum. Electron., vol. 63, no. 4, pp. 467-476, 2017.

[6] A. Sengupta and M. Rathor, “IP core steganography for protecting DSP kernels used in CE systems,” IEEE Trans. Consum. Electron., vol. 65, no. 4, pp. 506-515, 2019.

REFERENCE

[7] F. Koushanfar, I. Hong, M. Potkonjak, “Behavioral synthesis techniques for intellectual property protection,” ACM Trans. Design Autom. Electron. Syst. 10 (3) (2005) 523–545.

[8] A. Sengupta “Frontiers in Securing IP Cores - Forensic detective control and obfuscation techniques,” The Institute of Engineering and Technology (IET), 2020, ISBN: 1-83953-031-6.

[9] L. Li and A. Orailoglu, "Thwarting Reverse Engineering Attacks through Keyless Logic Obfuscation," IEEE 41st VLSI Test Symposium (VTS), San Diego, CA, USA, 2023, pp. 1-6.

[10] A. Sengupta and R. Chaurasia, “Securing IP Cores for DSP Applications Using Structural Obfuscation and Chromosomal DNA Impression,” IEEE Access, vol. 10, pp. 50903-50913, 2022.

[11] A. Sengupta, and D. Roy, “Protecting IP core during architectural synthesis using HLT-based obfuscation,” Electron. Lett., 2017, 53: 849-851.

[12] S. Amir, B. Shakya, X. Xu, Y. Jin, S. Bhunia, M. Tehranipoor & D. Forte, “Development and Evaluation of Hardware Obfuscation Benchmarks,” J Hardw Syst Secur 2, 142–161, 2018.

Thank You!