# Tech Talk Series @ PIEMR Indore

April 20, 2022

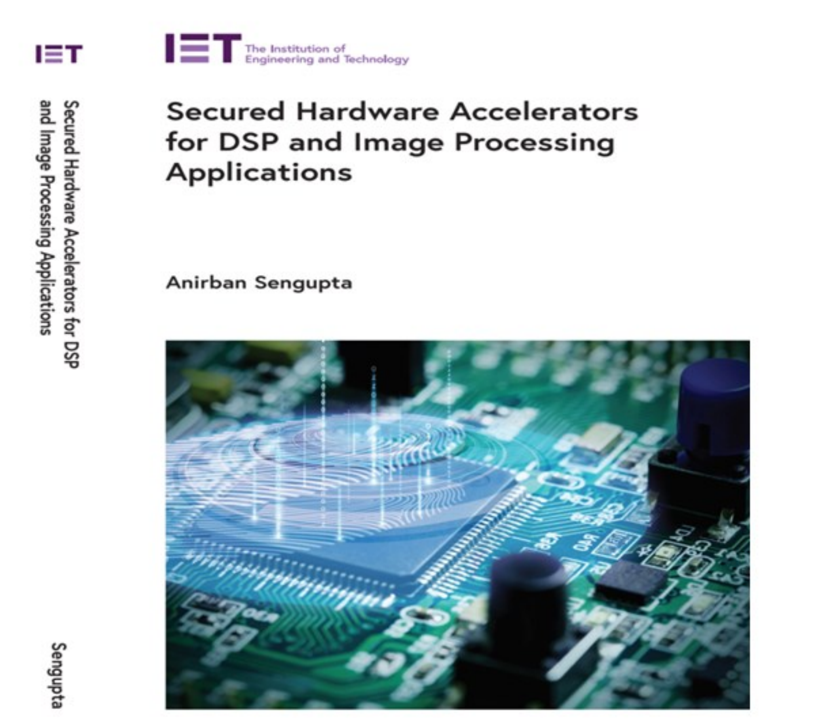
# Hardware based Cybersecurity for Image Processing and DSP

**DR. ANIRBAN SENGUPTA, ASSOC. PROFESSOR, Computer Science and Engineering, Indian Institute of Technology Indore**
**Ph.D (Canada), FIET (UK), FBCS, FIETE, P.Eng (Canada), SMIEEE**

**IEEE Distinguished Lecturer** (IEEE Consumer Electronics Society)
**IEEE Distinguished Visitor** (IEEE Computer Society)
**Ex-Officio - Board of Governors**, IEEE Consumer Electronics Society
**Former Chairman**, IEEE Computer Society Technical Committee on VLSI
**Chairman & Founder**, IEEE Consumer Electronics Society Bombay Chapter
**Chairman**, IEEE Consumer Electronics Society Technical Stream Committee - Security and Privacy
**Deputy Editor-in-Chief**, IET Computers & Digital Techniques,
**Former Editor-in-Chief**, IEEE VLSI Circuits and Systems Letter
**Associate Editor** - IEEE Transactions on VLSI Systems, IEEE Transactions on Aerospace and Electronic Systems, IEEE Transactions on Consumer Electronics, IEEE Letters of the Computer Society, IEEE Canadian Journal of Electrical and Computer Engineering
**Former Editorial Board Member** - IEEE Access, IEEE Consumer Electronics Magazine, IET Computers and Digital Techniques, Elsevier Microelectronics Journal
**General Chair**, 37th IEEE International Conference on Consumer Electronics (ICCE), Las Vegas
**General Chair**, 23rd International Symposium on VLSI Design and Test (VDAT-2019), India
**Executive Committee**, IEEE International Conference on Consumer Electronics (ICCE) - Berlin and Las Vegas
**IEEE Distinguished Lecturer Nominations Committee,** IEEE CE Society

*Email: asengupt@iiti.ac.in*
*Web: http://www.anirban-sengupta.com*

BOOKS


IET

Secured Hardware Accelerators for DSP and Image Processing Applications

Sengupta

IET The Institution of Engineering and Technology

## Secured Hardware Accelerators for DSP and Image Processing Applications

Anirban Sengupta

IET

Frontiers in Securing IP Cores

Forensic detective control and obfuscation techniques

Sengupta

IET The Institution of Engineering and Technology

## Frontiers in Securing IP Cores

Forensic detective control and obfuscation techniques

Anirban Sengupta

IET

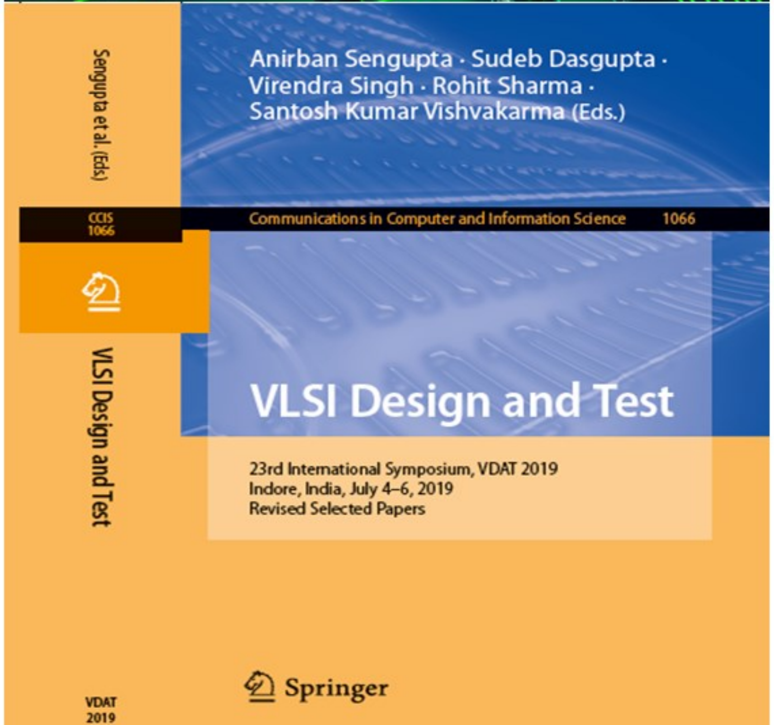
IP Core Protection and Hardware-Assisted Security for Consumer Electronics

Sengupta and Mohanty

IET The Institution of Engineering and Technology

## IP Core Protection and Hardware-Assisted Security for Consumer Electronics

Anirban Sengupta and Saraju P. Mohanty


Sengupta et al. (Eds.)

CCIS 1066

VLSI Design and Test

VDAT 2019

Anirban Sengupta · Sudeb Dasgupta · Virendra Singh · Rohit Sharma · Santosh Kumar Vishvakarma (Eds.)

Communications in Computer and Information Science 1066

## VLSI Design and Test

23rd International Symposium, VDAT 2019
Indore, India, July 4–6, 2019
Revised Selected Papers

Springer

DR. ANIRBAN SENGUPTA

# Introduction

- Hardware Security and Intellectual Property (IP) Core protection is an emerging area of research for semiconductor community that focusses on protecting designs against standard threats such as reverse engineering, counterfeit, forgery, malicious hardware modification etc.

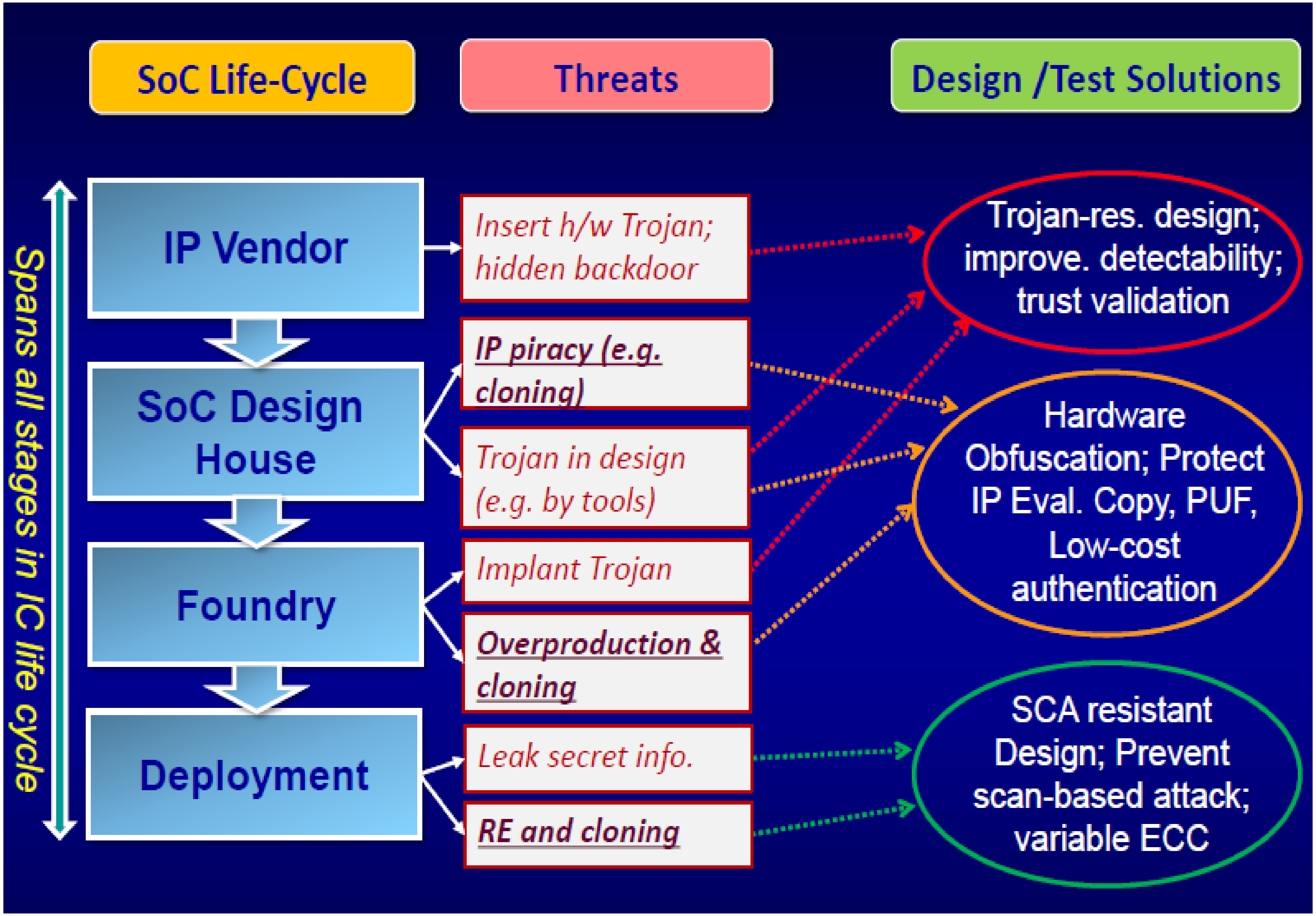
- Hardware security is broadly classified into two types: (a) authentication based approaches (b) obfuscation based approaches.

- The second type of hardware security approach i.e. obfuscation can again be further sub-divided into two types: (i) structural obfuscation (ii) functional obfuscation. Structural obfuscation transforms a design into one that is functionally equivalent to the original but is significantly more difficult to reverse engineer (RE), while the second one is active protection type that locks the design through a secret key.

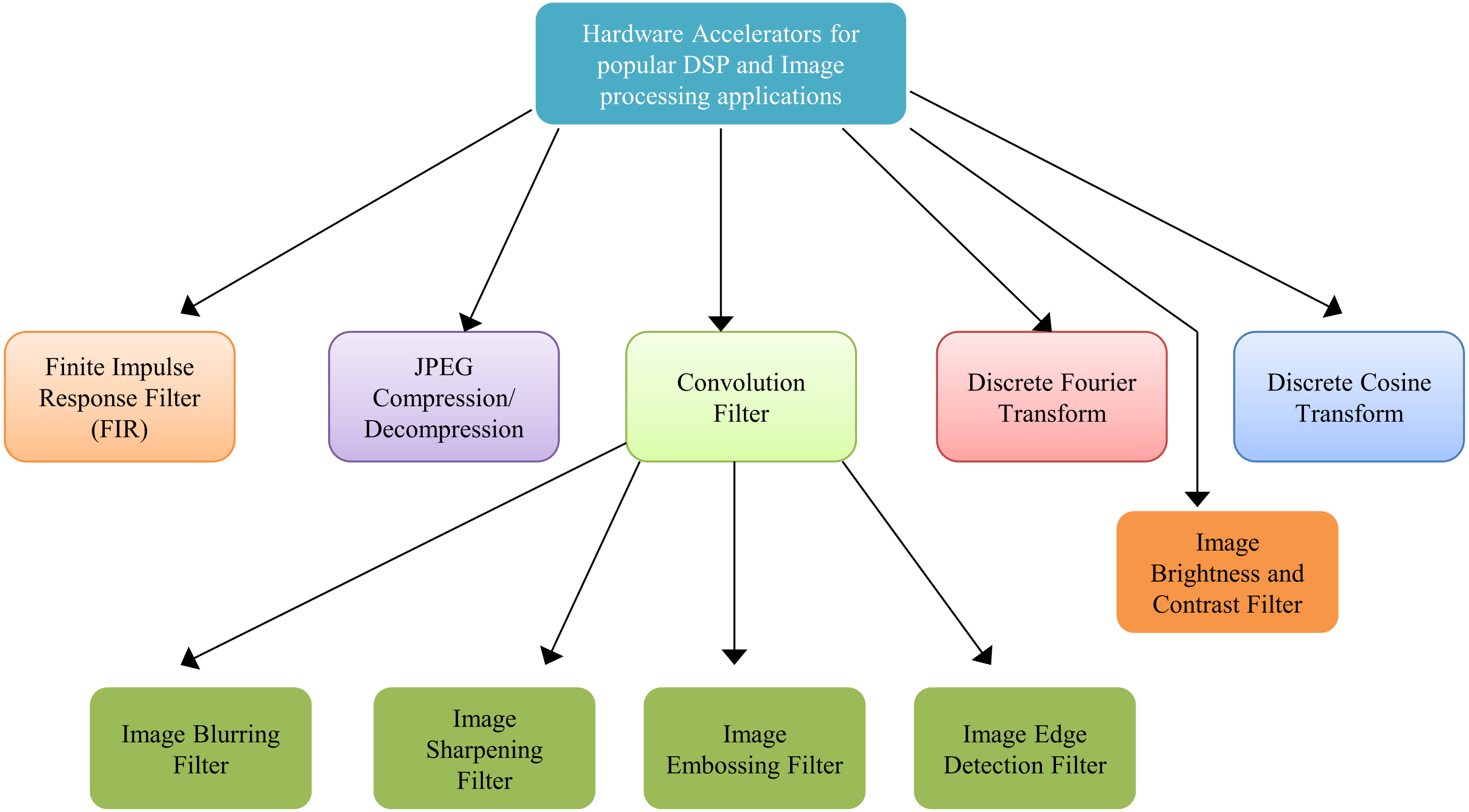
**Anirban Sengupta** "Frontiers in Securing IP Cores - Forensic detective control and obfuscation techniques", **The Institute of Engineering and Technology (IET),** 2020, ISBN-10: 1-83953-031-6, ISBN-13: 978-1-83953-031-9

Anirban Sengupta, Mahendra Rathor "Protecting DSP Kernels using Robust Hologram based Obfuscation", **IEEE Transactions on Consumer Electronics**, 2019

# IP Core Protection and Hardware Security

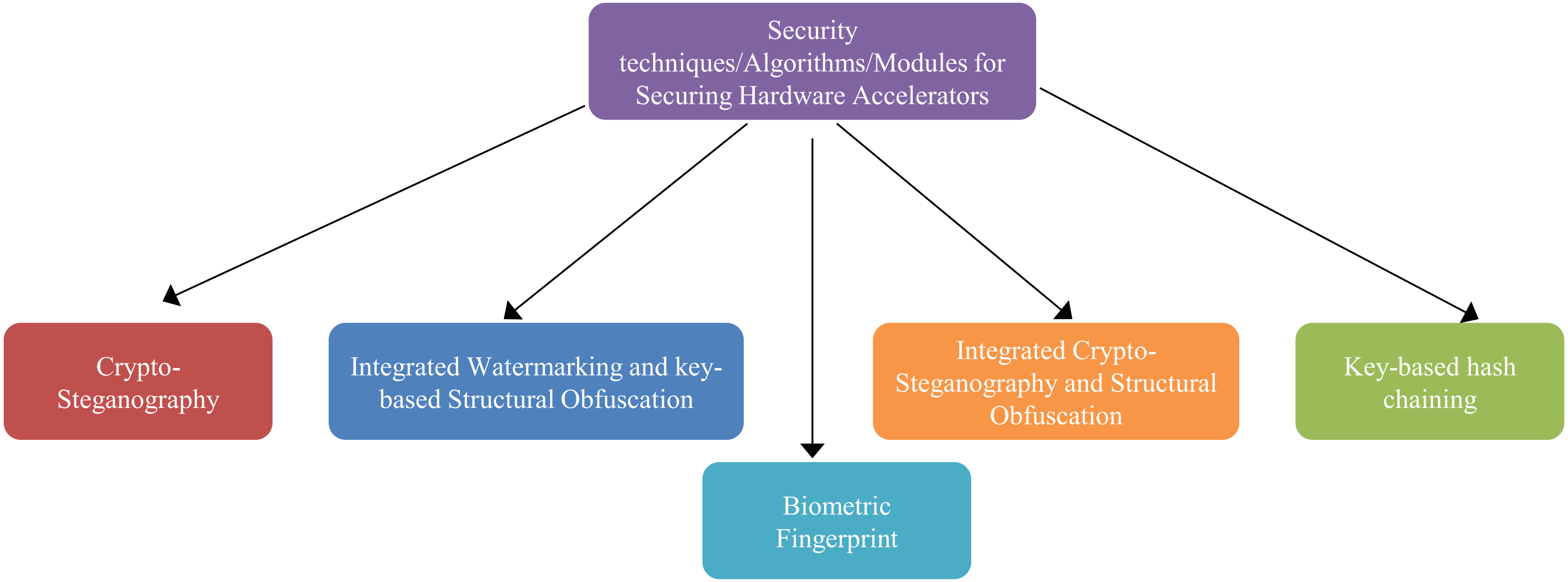

SoC Life-Cycle
Threats
Design /Test Solutions
Spans all stages in IC life cycle
IP Vendor
SoC Design House
Foundry
Deployment
Insert h/w Trojan; hidden backdoor
IP piracy (e.g. cloning)
Trojan in design (e.g. by tools)
Implant Trojan
Overproduction & cloning
Leak secret info.
RE and cloning
Trojan-res. design; improve. detectability; trust validation
Hardware Obfuscation; Protect IP Eval. Copy, PUF, Low-cost authentication
SCA resistant Design; Prevent scan-based attack; variable ECC

# Hardware accelerators: Example


Hardware Accelerators for popular DSP and Image processing applications
Finite Impulse Response Filter (FIR)
JPEG Compression/ Decompression
Convolution Filter
Discrete Fourier Transform
Discrete Cosine Transform
Image Brightness and Contrast Filter
Image Blurring Filter
Image Sharpening Filter
Image Embossing Filter
Image Edge Detection Filter

**Anirban Sengupta "Frontiers in Securing IP Cores - Forensic detective control and obfuscation techniques", The Institute of Engineering and Technology (IET),** 2020, ISBN-10: 1-83953-031-6, ISBN-13: 978-1-83953-031-9

# Hardware security techniques for securing hardware accelerators



Security techniques/Algorithms/Modules for Securing Hardware Accelerators

Crypto-Steganography

Integrated Watermarking and key-based Structural Obfuscation

Biometric Fingerprint

Integrated Crypto-Steganography and Structural Obfuscation

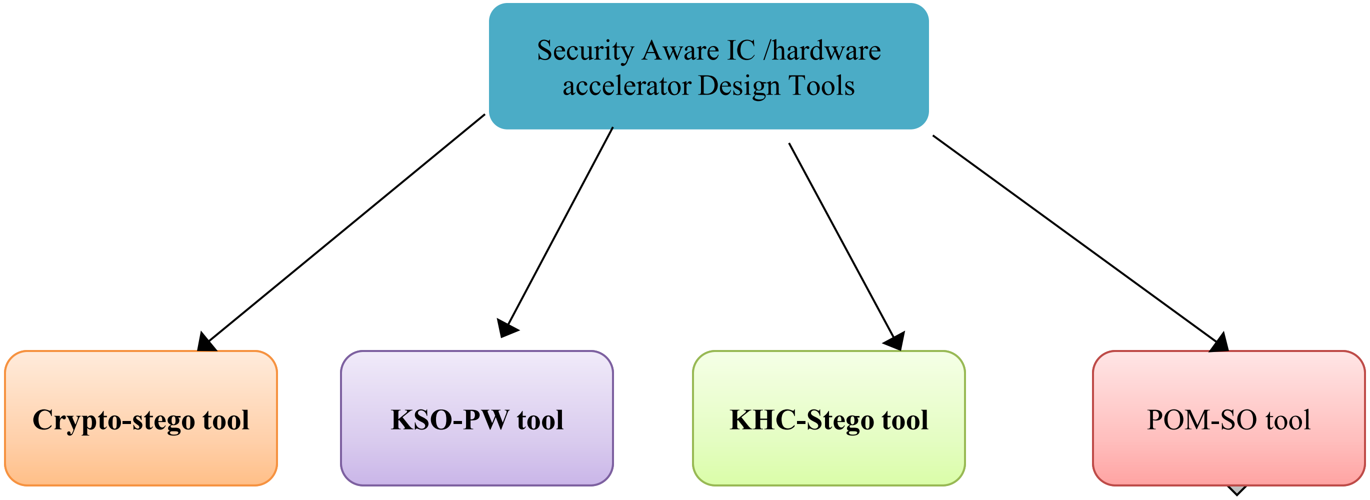
Key-based hash chaining

**Anirban Sengupta "Frontiers in Securing IP Cores - Forensic detective control and obfuscation techniques", The Institute of Engineering and Technology (IET),** 2020, ISBN-10: 1-83953-031-6, ISBN-13: 978-1-83953-031-9

# Our in-house hardware security tools for designing secured accelerators

Released publicly from our group in Sep 2020 !

http://www.anirban-sengupta.com/Hardware_Security_Tools.php



Security Aware IC /hardware accelerator Design Tools

**Crypto-stego tool**

**KSO-PW tool**

**KHC-Stego tool**

POM-SO tool

Crypto-steganography tool for DSP hardware accelerators

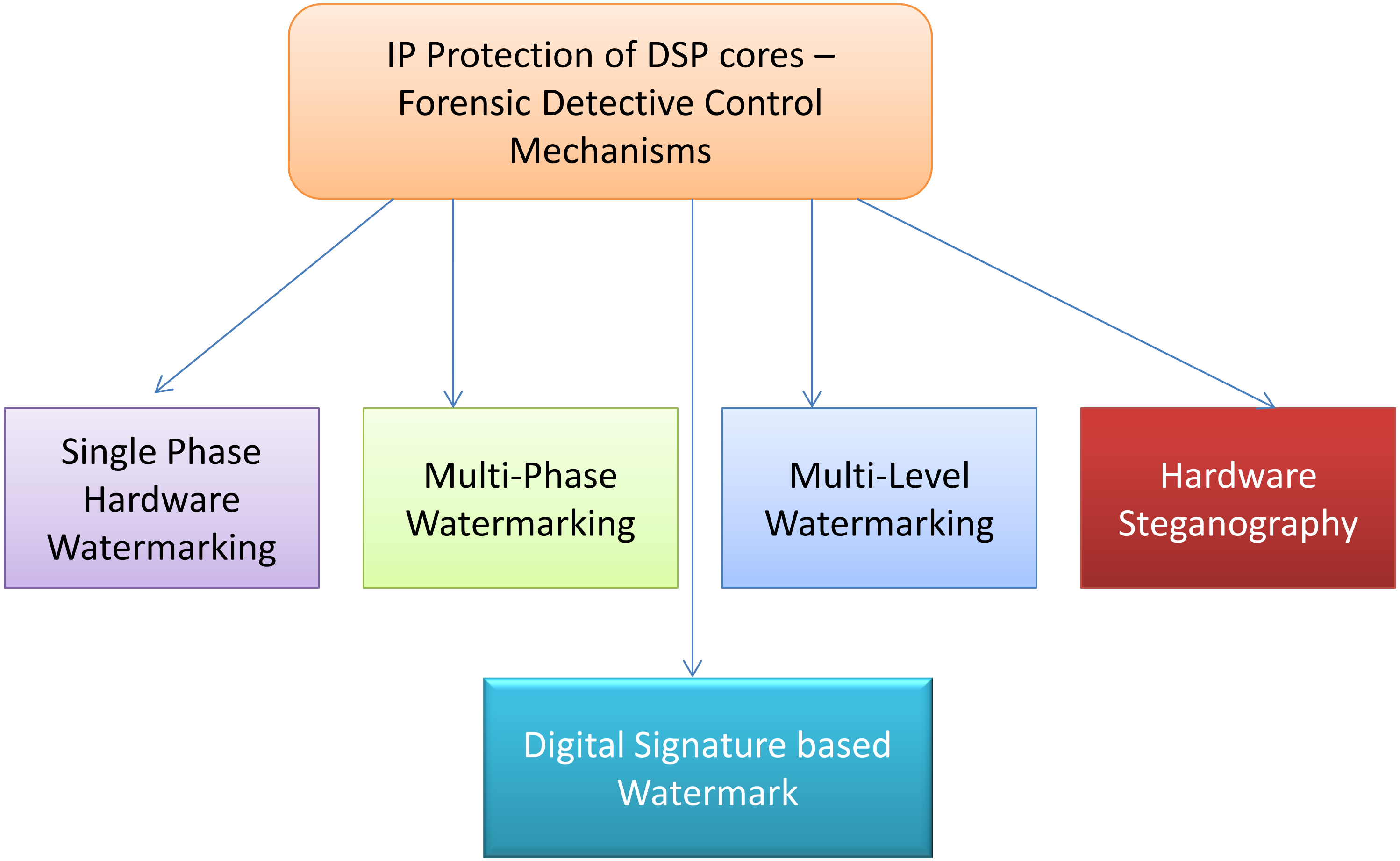
Key-driven structural obfuscation and physical level watermarking tool

Key-triggered Hash-chaining Driven Steganography tool

Pseudo operation mixing based structural obfuscation tool

**Anirban Sengupta "Frontiers in Securing IP Cores - Forensic detective control and obfuscation techniques", The Institute of Engineering and Technology (IET),** 2020, ISBN-10: 1-83953-031-6, ISBN-13: 978-1-83953-031-9

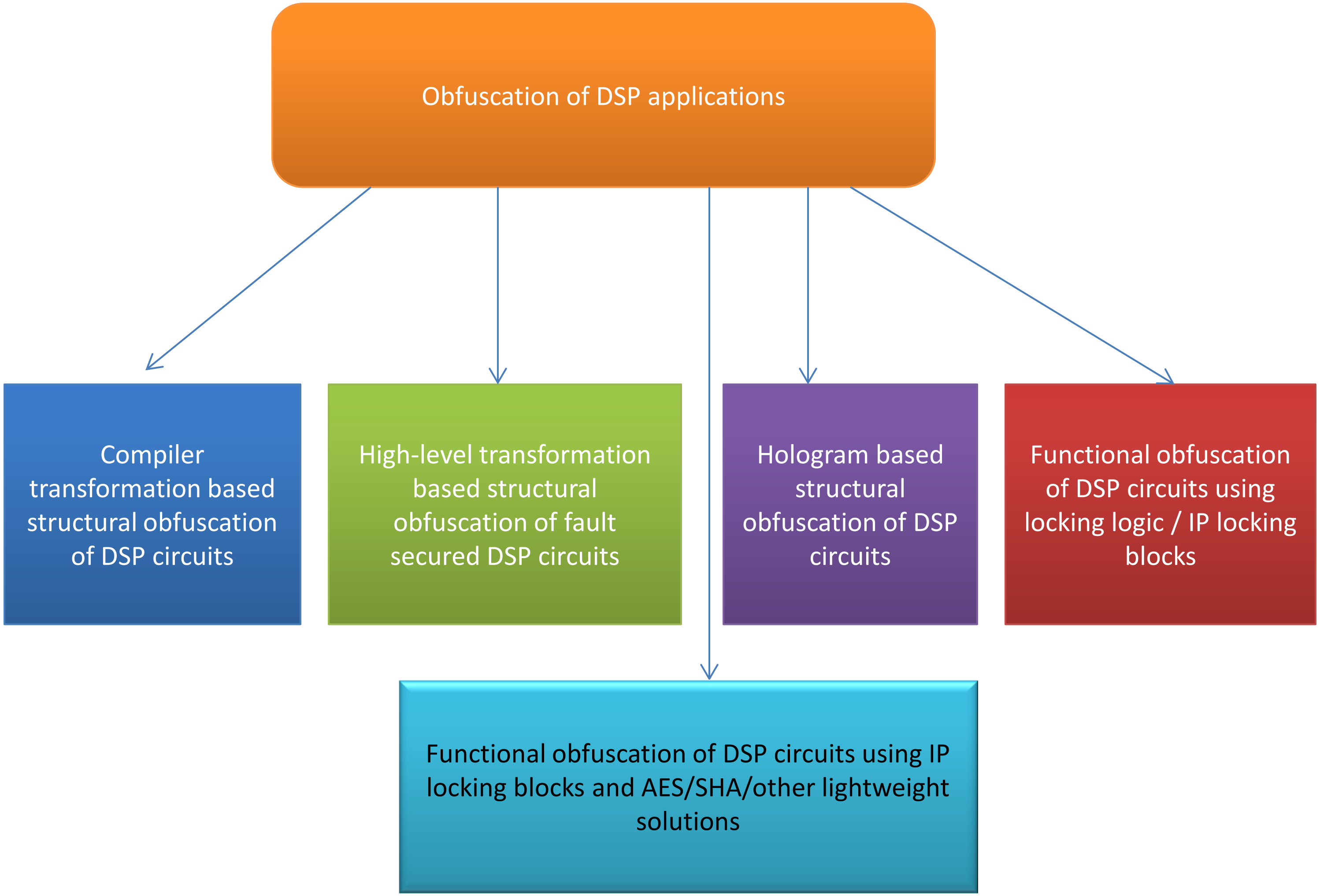
# IP Protection of DSP cores – Forensic Detective Control Mechanisms



IP Protection of DSP cores – Forensic Detective Control Mechanisms
Single Phase Hardware Watermarking
Multi-Phase Watermarking
Multi-Level Watermarking
Hardware Steganography
Digital Signature based Watermark

**Anirban Sengupta,** Saraju P. Mohanty "**IP Core Protection and Hardware-Assisted Security for Consumer Electronics**", **The Institute of Engineering and Technology (IET),** 2019, Book ISBN: 978-1-78561-799-7, e-ISBN: 978-1-78561-800-0

# Hardware Security of DSP applications using Obfuscation


Obfuscation of DSP applications

Compiler transformation based structural obfuscation of DSP circuits

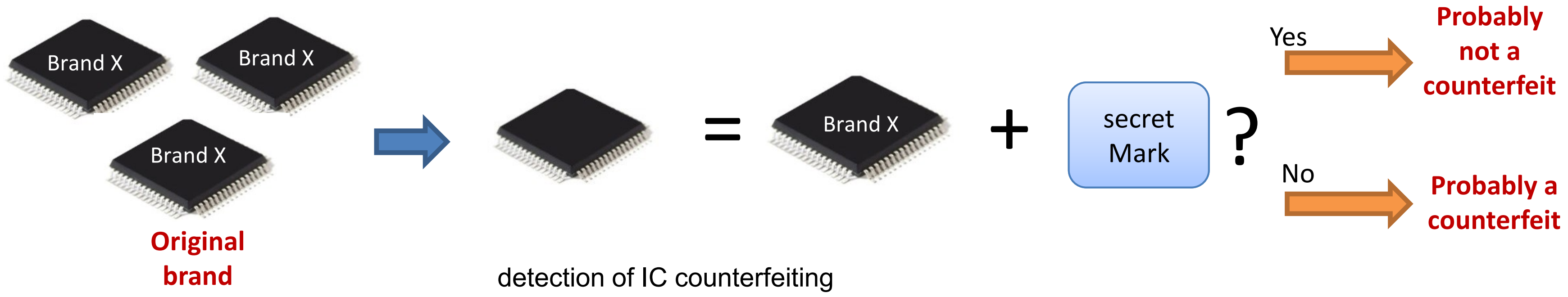
High-level transformation based structural obfuscation of fault secured DSP circuits

Hologram based structural obfuscation of DSP circuits

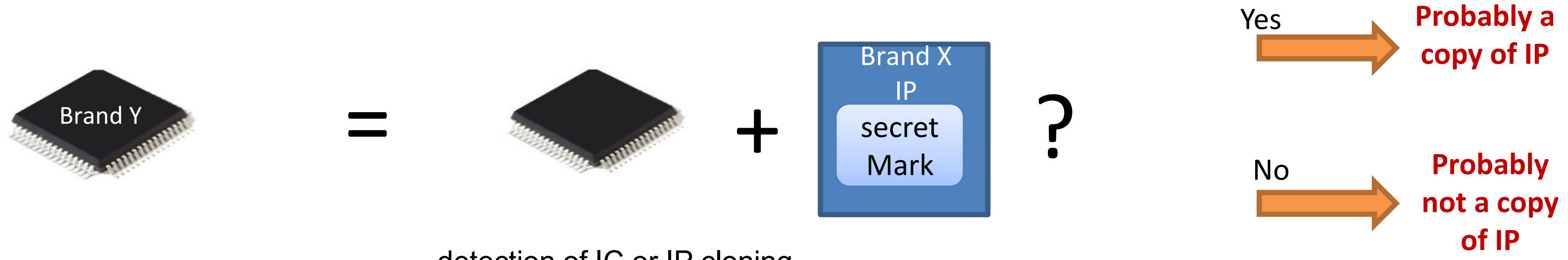
Functional obfuscation of DSP circuits using locking logic / IP locking blocks

Functional obfuscation of DSP circuits using IP locking blocks and AES/SHA/other lightweight solutions

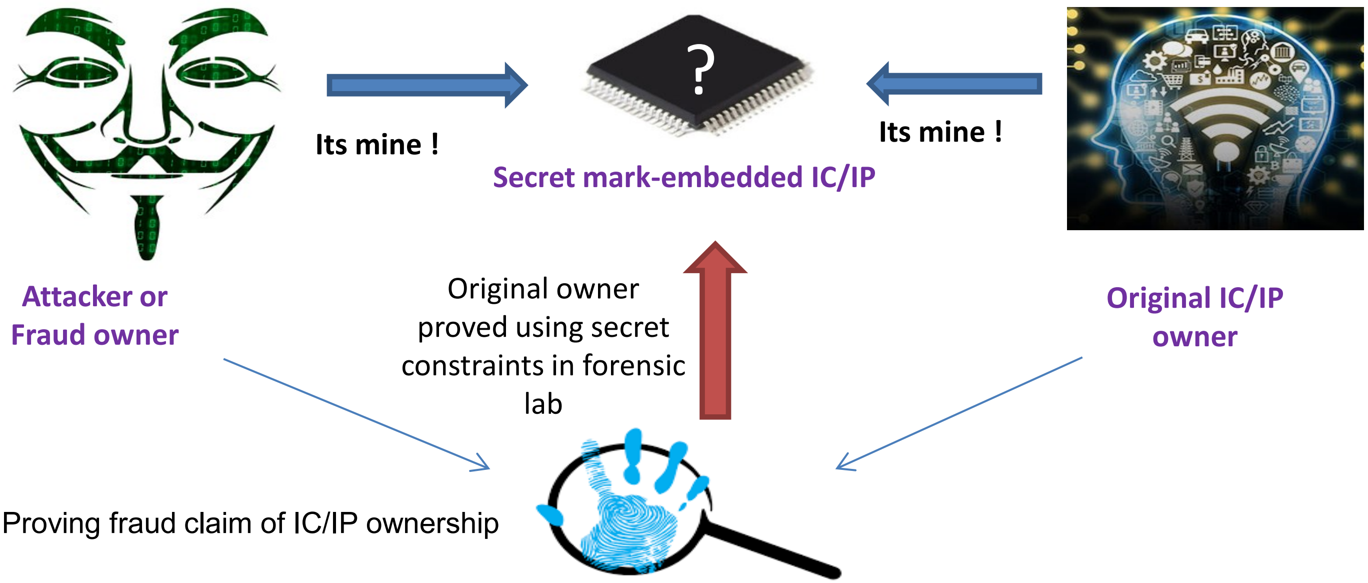
**Anirban Sengupta "Frontiers in Securing IP Cores - Forensic detective control and obfuscation techniques", The Institute of Engineering and Technology (IET),** 2020, ISBN-10: 1-83953-031-6, ISBN-13: 978-1-83953-031-9



detection of IC counterfeiting



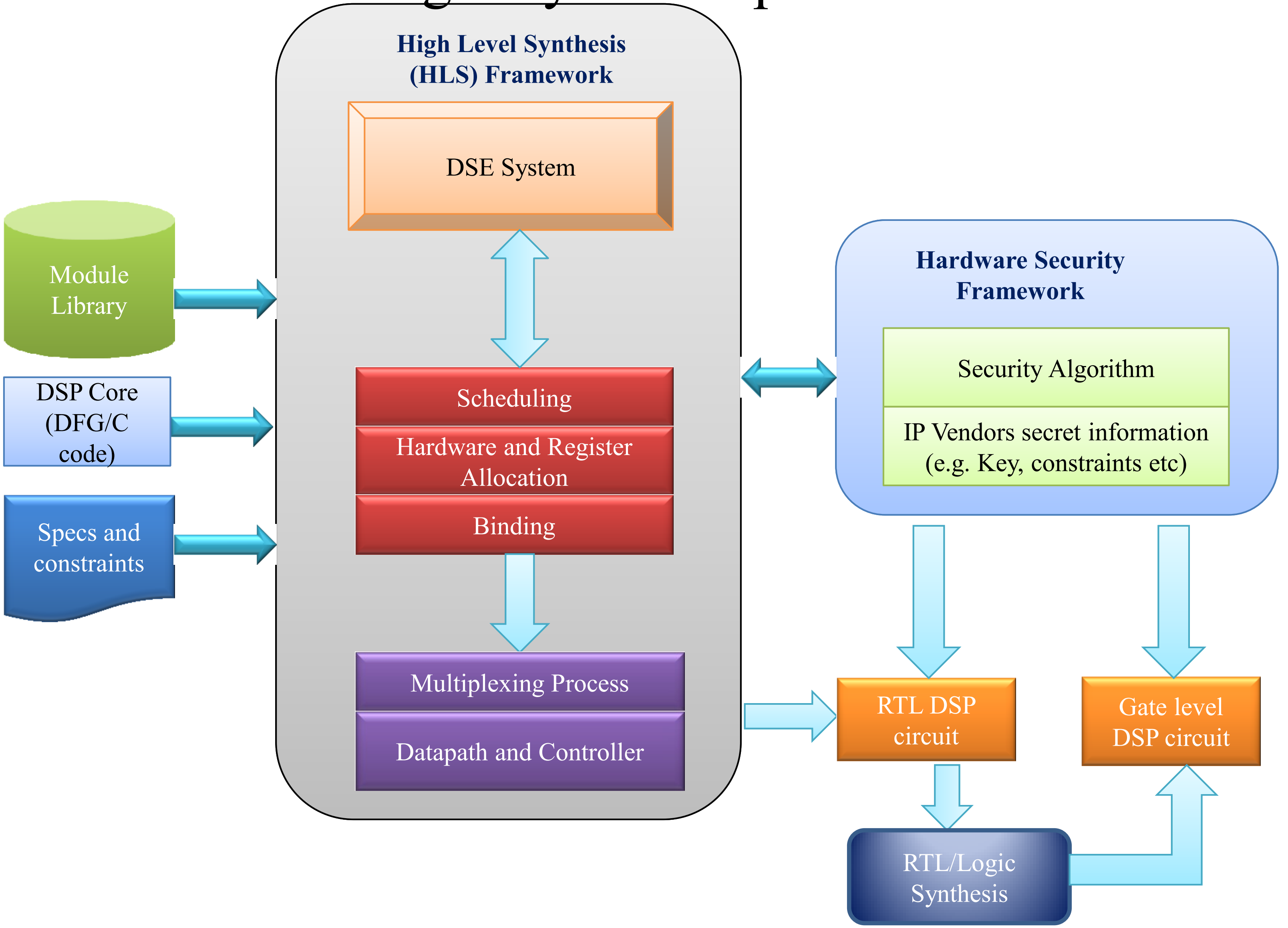
detection of IC or IP cloning


?
Its mine !
Its mine !
Secret mark-embedded IC/IP
Attacker or Fraud owner
Original owner proved using secret constraints in forensic lab
Original IC/IP owner

Proving fraud claim of IC/IP ownership

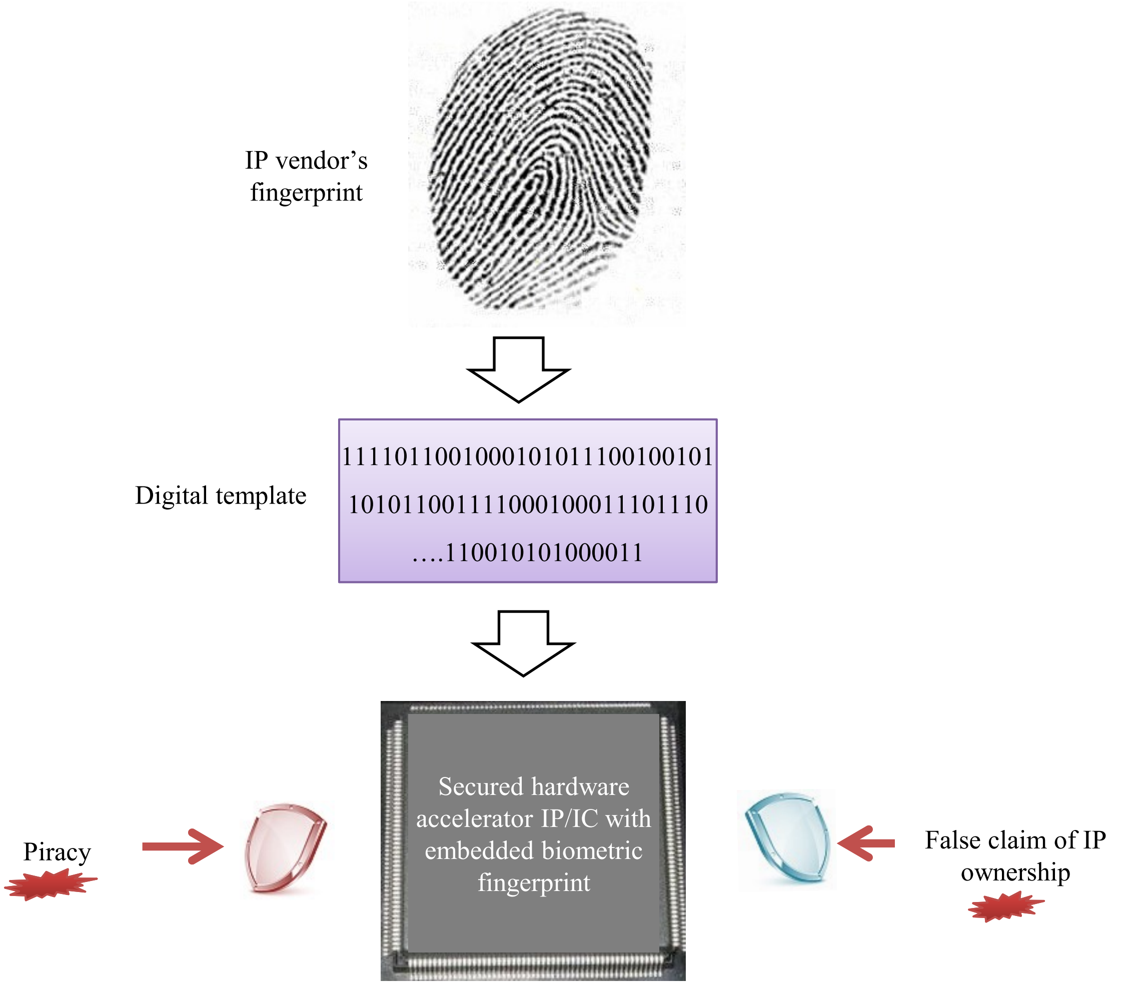
**Anirban Sengupta**, Mahendra Rathor "IP Core Steganography for Protecting DSP Kernels used in CE Systems", **IEEE Transactions on Consumer Electronics (TCE)** , Volume: 65 , Issue: 4 , Nov. 2019, pp. 506 - 515

# Hardware Security Algorithms integrated with HLS and Logic Synthesis phases



**Anirban Sengupta "Frontiers in Securing IP Cores - Forensic detective control and obfuscation techniques", The Institute of Engineering and Technology (IET),** 2020, ISBN-10: 1-83953-031-6, ISBN-13: 978-1-83953-031-9

# Securing hardware accelerators using biometric fingerprinting: Forensics



IP vendor's fingerprint

Digital template

1111011001000101011100100101
101011001111000100011101110
….110010101000011

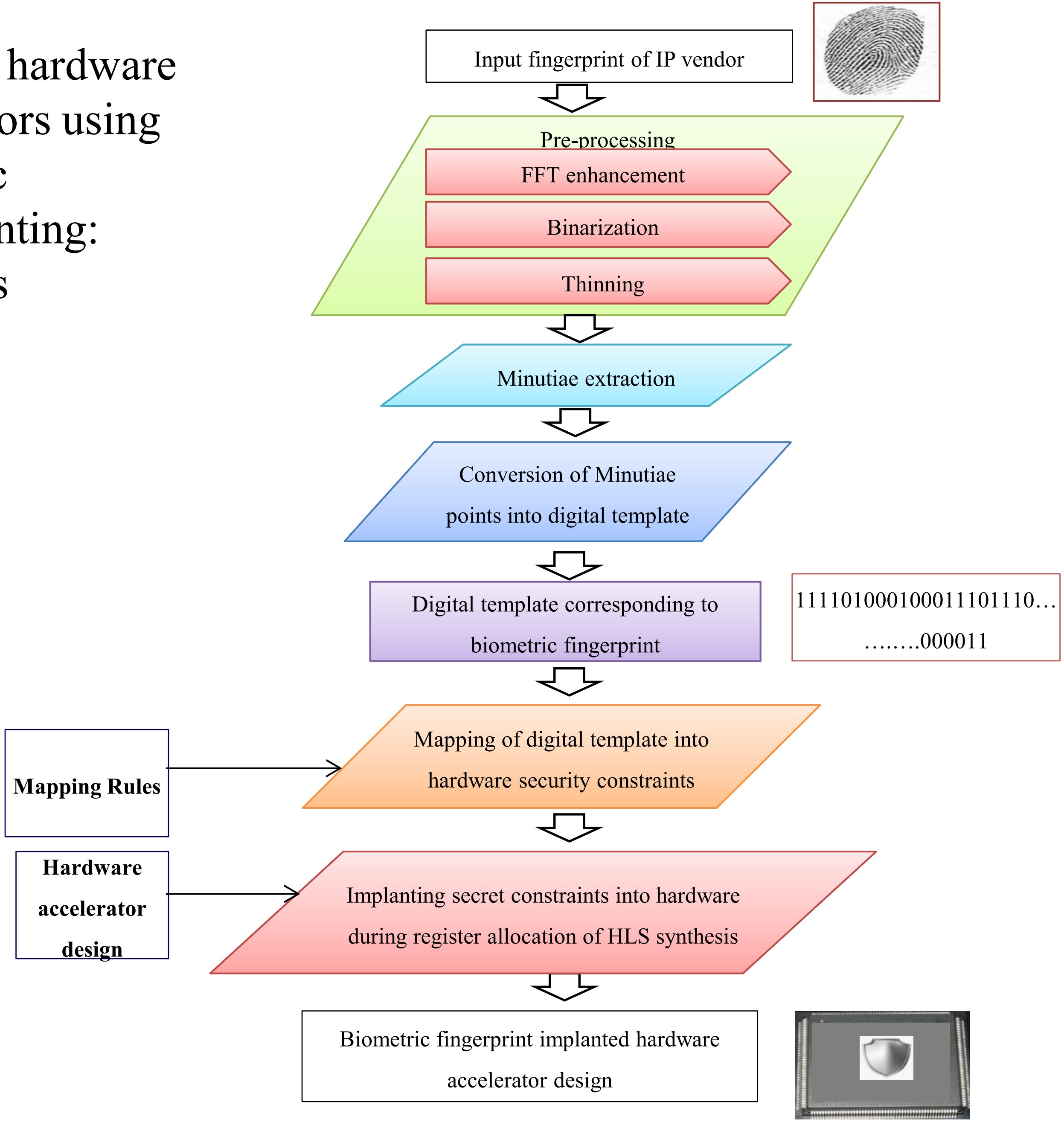
Secured hardware accelerator IP/IC with embedded biometric fingerprint

Piracy

False claim of IP ownership

**Anirban Sengupta**, Mahendra Rathor "Securing Hardware Accelerators for CE Systems using Biometric Fingerprinting", **IEEE Transactions on Very Large Scale Integration Systems (TVLSI)** , Accepted, 2020

# Securing hardware accelerators using biometric fingerprinting: Forensics



Input fingerprint of IP vendor

Pre-processing

FFT enhancement

Binarization

Thinning

Minutiae extraction

Conversion of Minutiae points into digital template

Digital template corresponding to biometric fingerprint

1111010001000111011110…

……..000011

Mapping Rules

Mapping of digital template into hardware security constraints

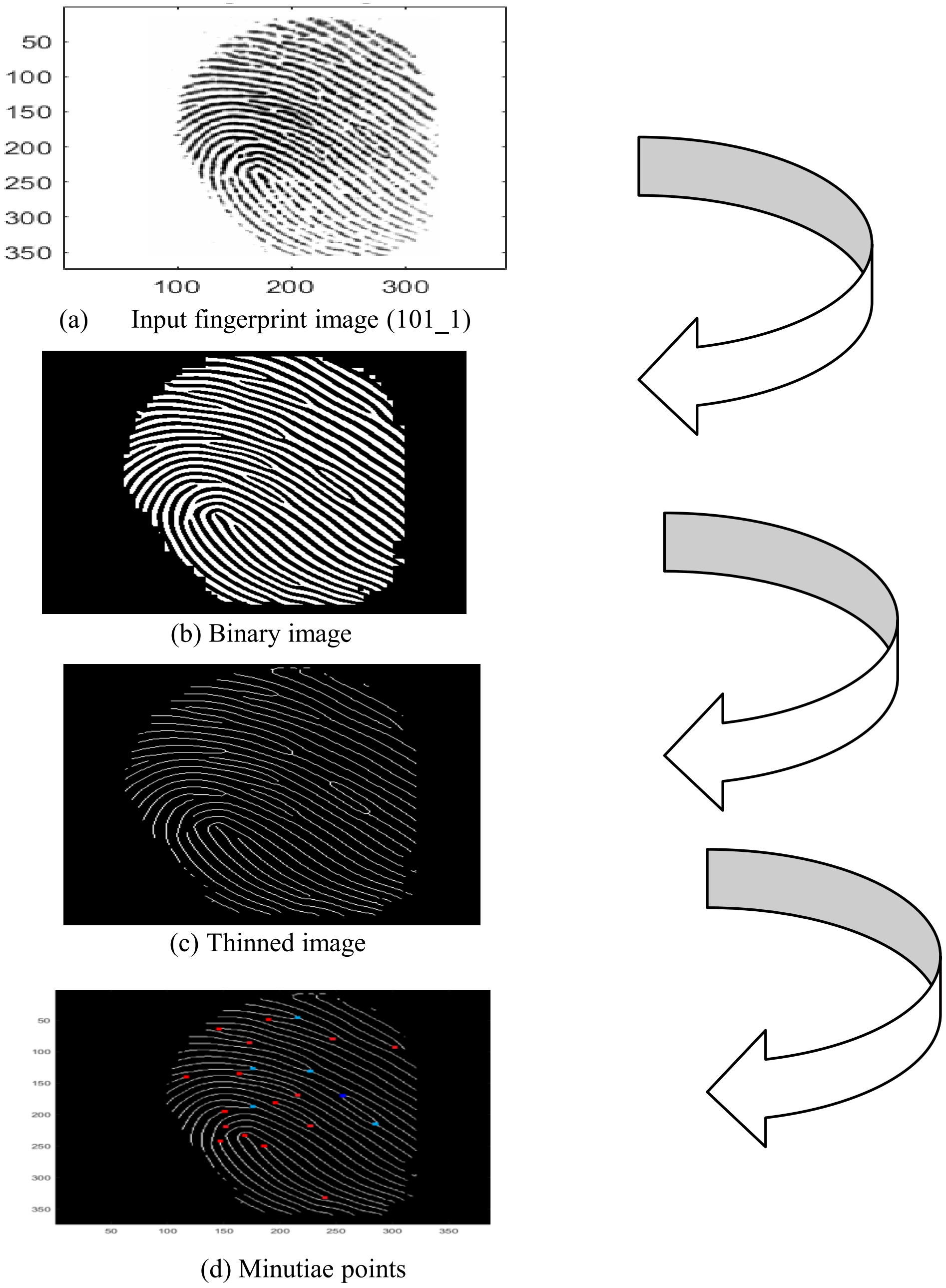
Hardware accelerator design

Implanting secret constraints into hardware during register allocation of HLS synthesis

Biometric fingerprint implanted hardware accelerator design

**Anirban Sengupta**, Mahendra Rathor "Securing Hardware Accelerators for CE Systems using Biometric Fingerprinting", **IEEE Transactions on Very Large Scale Integration Systems (TVLSI)** , Accepted, 2020

# Securing hardware accelerators using biometric fingerprinting: Forensics: An example



(a) Input fingerprint image (101_1)

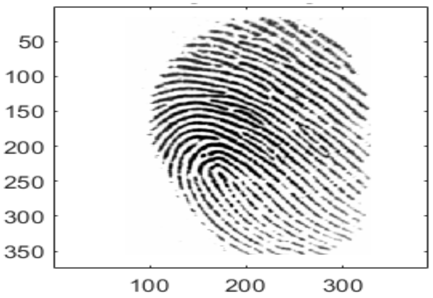
(b) Binary image

(c) Thinned image

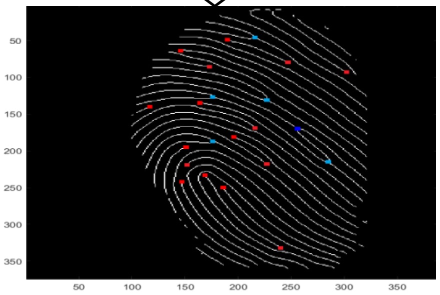
(d) Minutiae points

Minutiae points extraction flow (a) Captured fingerprint image (b) Binary fingerprint image post enhancement (c) Fingerprint image post applying thinning (d) Fingerprint image with minutiae points located

# Secret biometric constraints for various fingerprint images



Fingerprint image (101_1)



Minutiae points=22

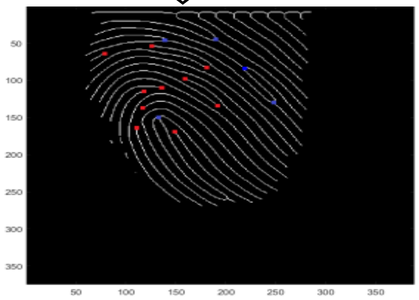
1101100010111011111101101111
10……….101001100111101

Corresponding Digital Template
(Total size =526 bits)

#0s= 224
#1s= 302

Secret biometric constraints
for IC/IP

Fingerprint image:101_2



Minutiae points=15

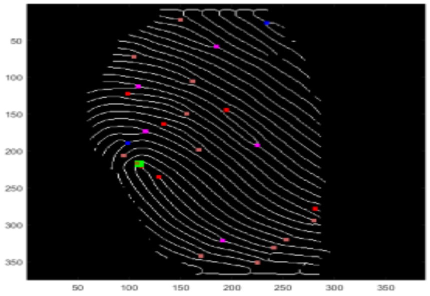
10111110101101111000101000
10111……..01001111101111

Corresponding Digital Template
(Total size =350 bits)

#0s= 148
#1s= 202

Secret biometric constraints
for IC/IP

Fingerprint image:101_8



Minutiae points=24

10010110101101101001110101
011010…..110101111111110

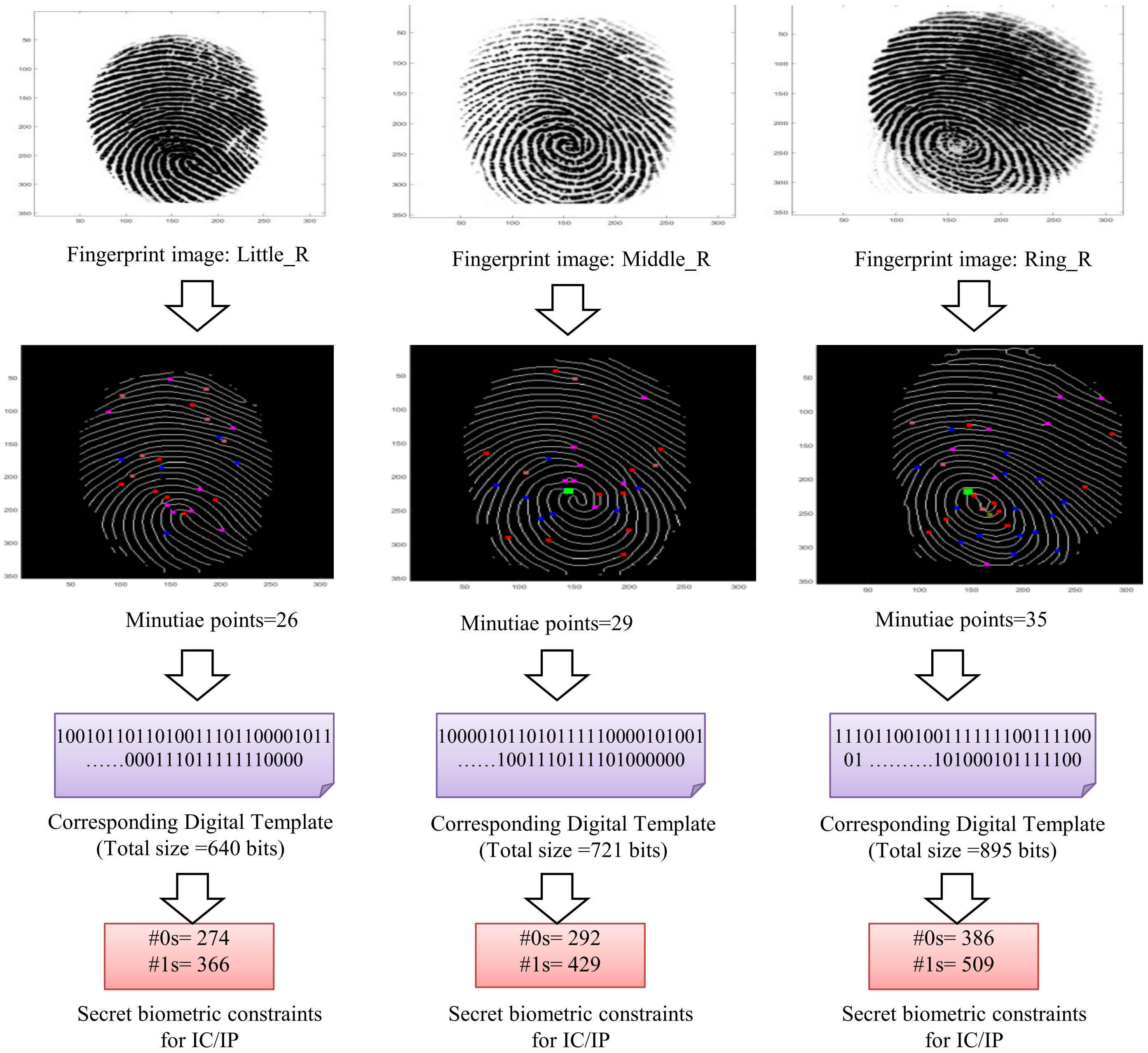
Corresponding Digital Template
(Total size =555 bits)

#0s= 242
#1s= 312

Secret biometric constraints
for IC/IP

**Anirban Sengupta**, Mahendra Rathor "Securing Hardware Accelerators for CE Systems using Biometric Fingerprinting", **IEEE Transactions on Very Large Scale Integration Systems (TVLSI)** , Accepted, 2020

# Secret biometric constraints for fingerprints of different fingers of an individual



Fingerprint image: Little_R

Minutiae points=26

1001011011010011101100001011
……0001110111111110000

Corresponding Digital Template
(Total size =640 bits)

#0s= 274
#1s= 366

Secret biometric constraints
for IC/IP

Fingerprint image: Middle_R

Minutiae points=29

1000010110101111110000101001
……1001110111101000000

Corresponding Digital Template
(Total size =721 bits)

#0s= 292
#1s= 429

Secret biometric constraints
for IC/IP

Fingerprint image: Ring_R

Minutiae points=35

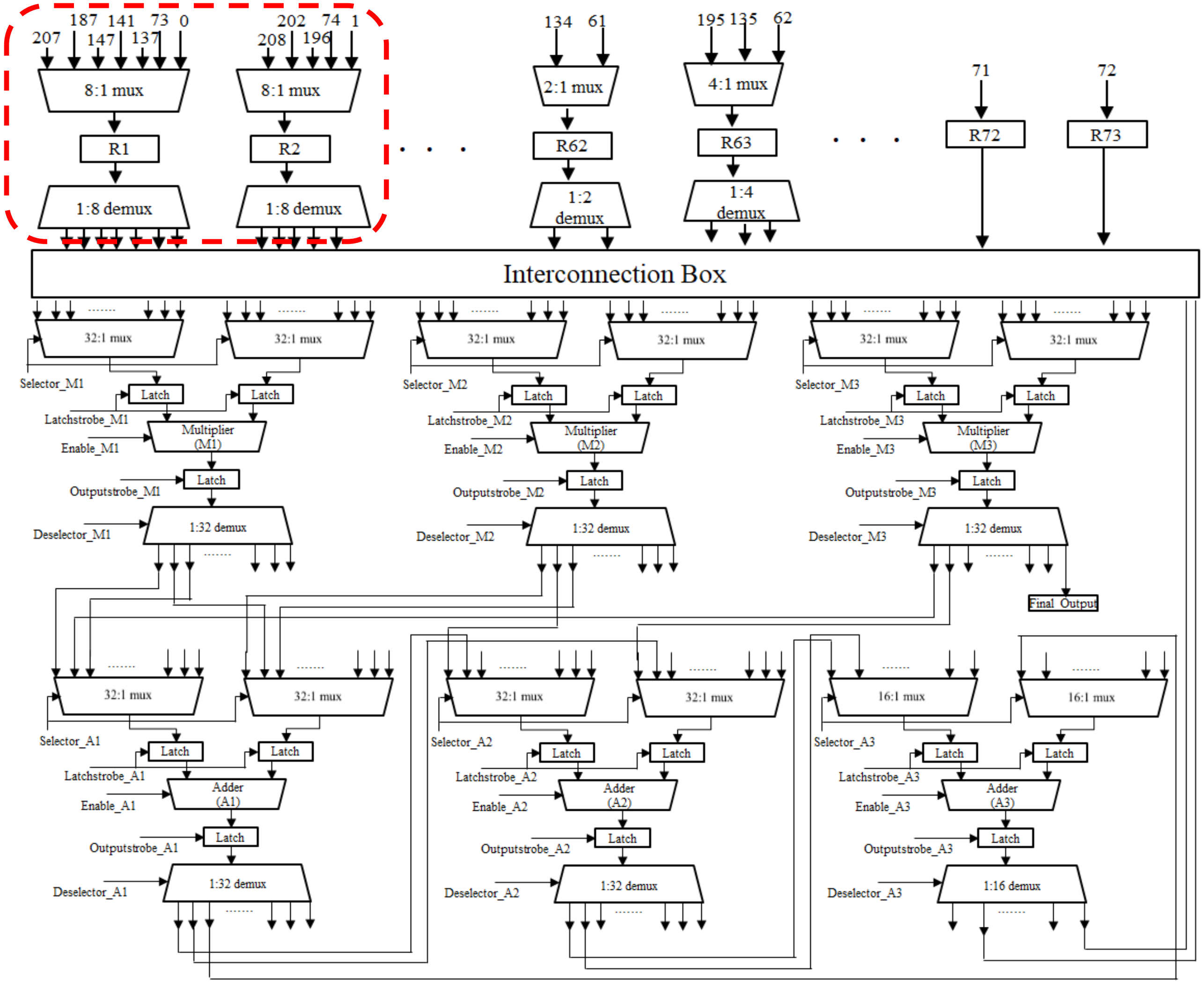
111011001001111111100111100
01 ……….101000101111100

Corresponding Digital Template
(Total size =895 bits)

#0s= 386
#1s= 509

Secret biometric constraints
for IC/IP

# Secured datapath of JPEG compression hardware accelerator implanted with biometric fingerprint



**Anirban Sengupta**, Mahendra Rathor "Securing Hardware Accelerators for CE Systems using Biometric Fingerprinting", **IEEE Transactions on Very Large Scale Integration Systems (TVLSI)** , Accepted, 2020

# Detecting Biometric Fingerprint in a hardware accelerator

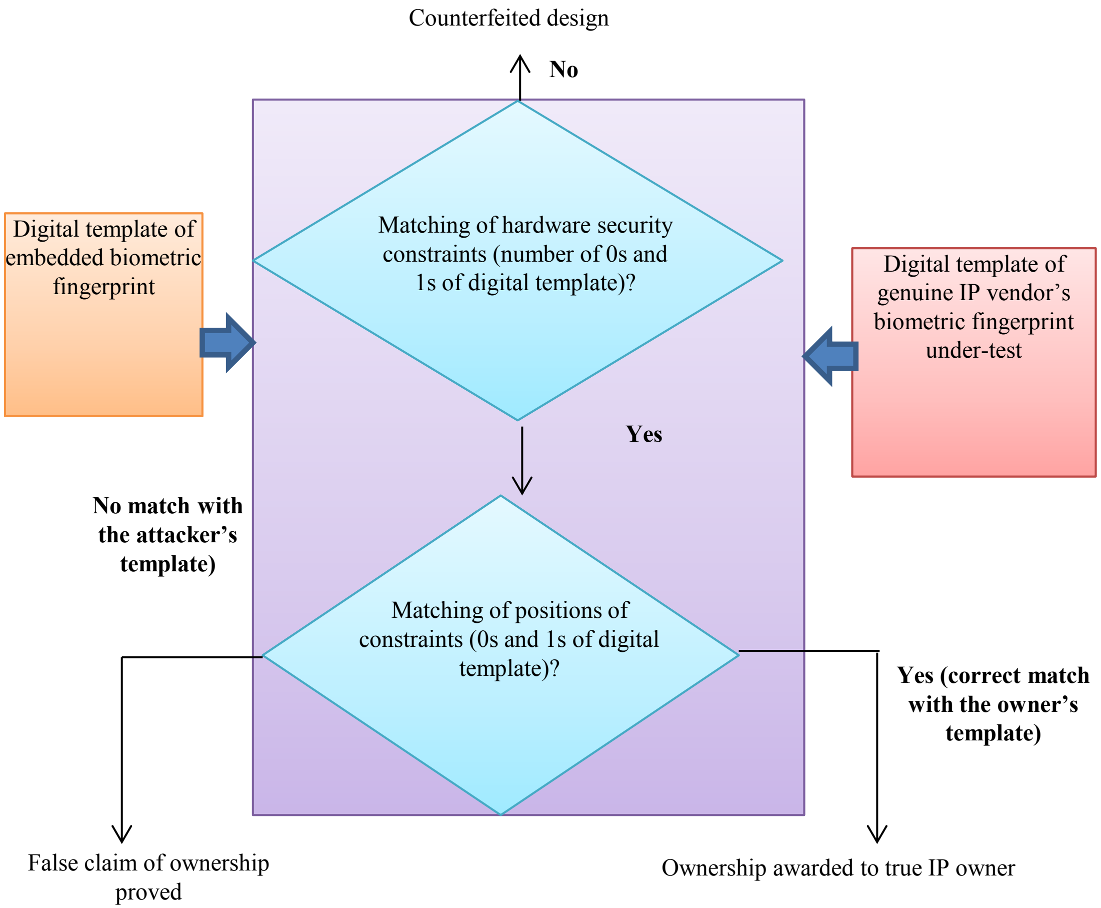
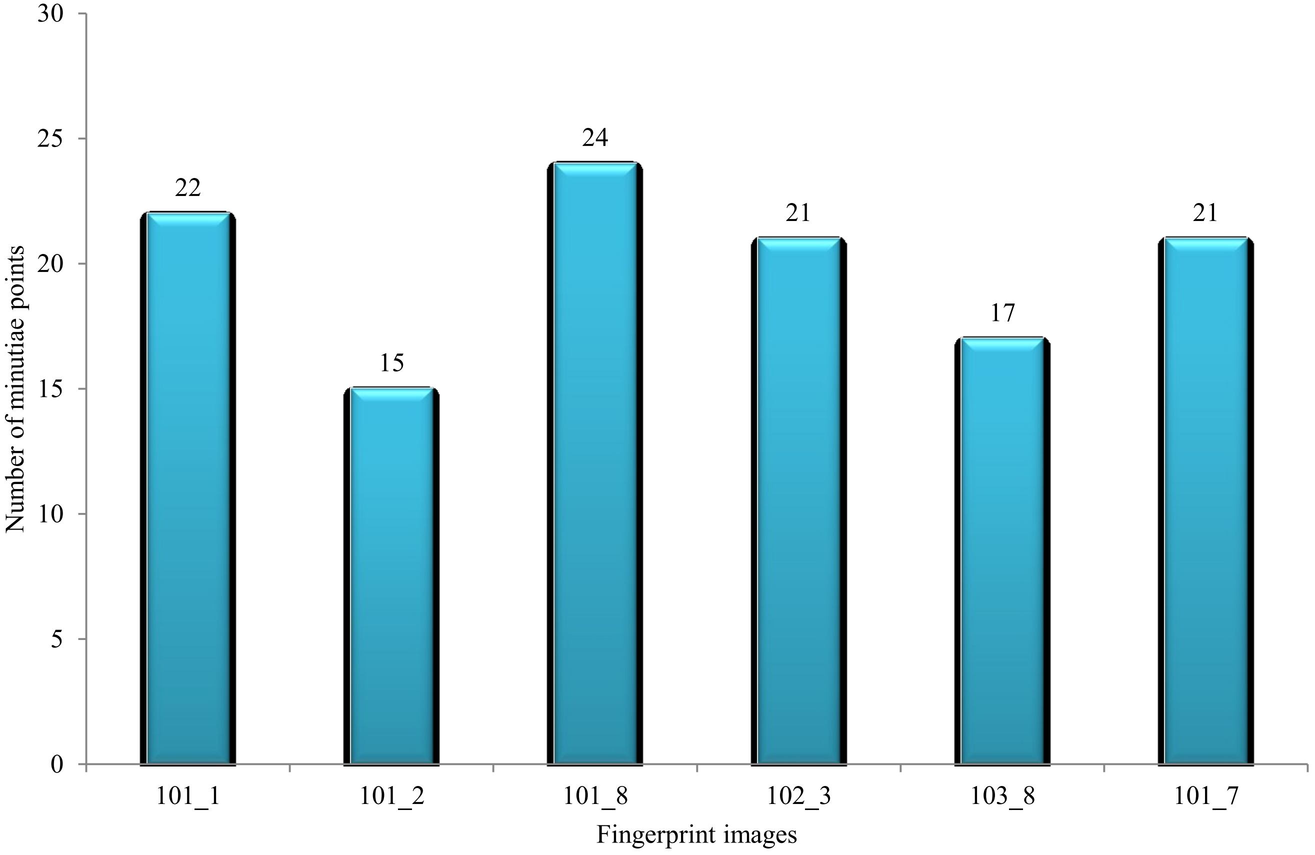

Counterfeited design
No
Digital template of embedded biometric fingerprint
Matching of hardware security constraints (number of 0s and 1s of digital template)?
Digital template of genuine IP vendor's biometric fingerprint under-test
Yes
No match with the attacker's template)
Matching of positions of constraints (0s and 1s of digital template)?
Yes (correct match with the owner's template)
False claim of ownership proved
Ownership awarded to true IP owner

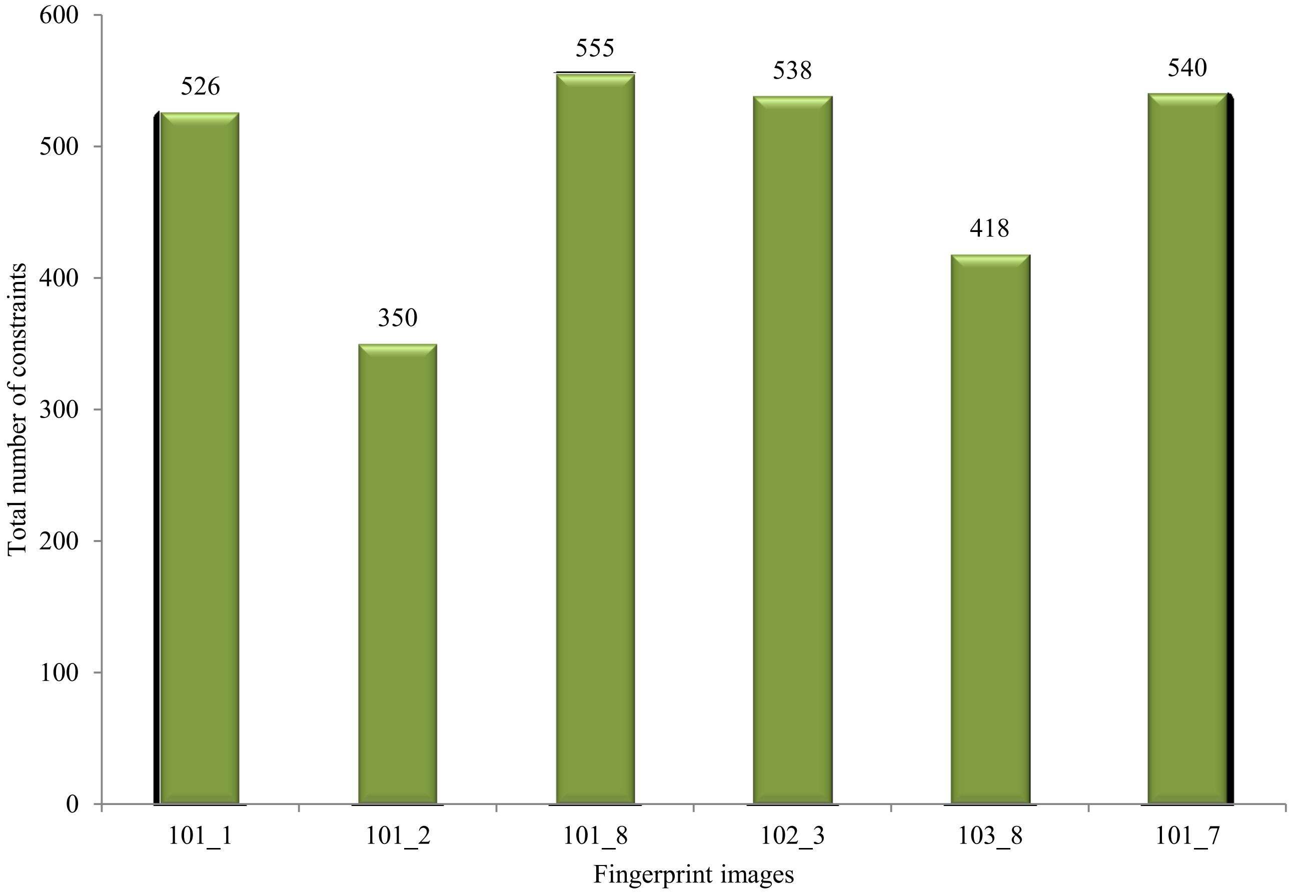
Fig. 9. Proving true IP ownership using proposed detection approach

**Anirban Sengupta**, Mahendra Rathor "Securing Hardware Accelerators for CE Systems using Biometric Fingerprinting", **IEEE Transactions on Very Large Scale Integration Systems (TVLSI)** , Accepted, 2020

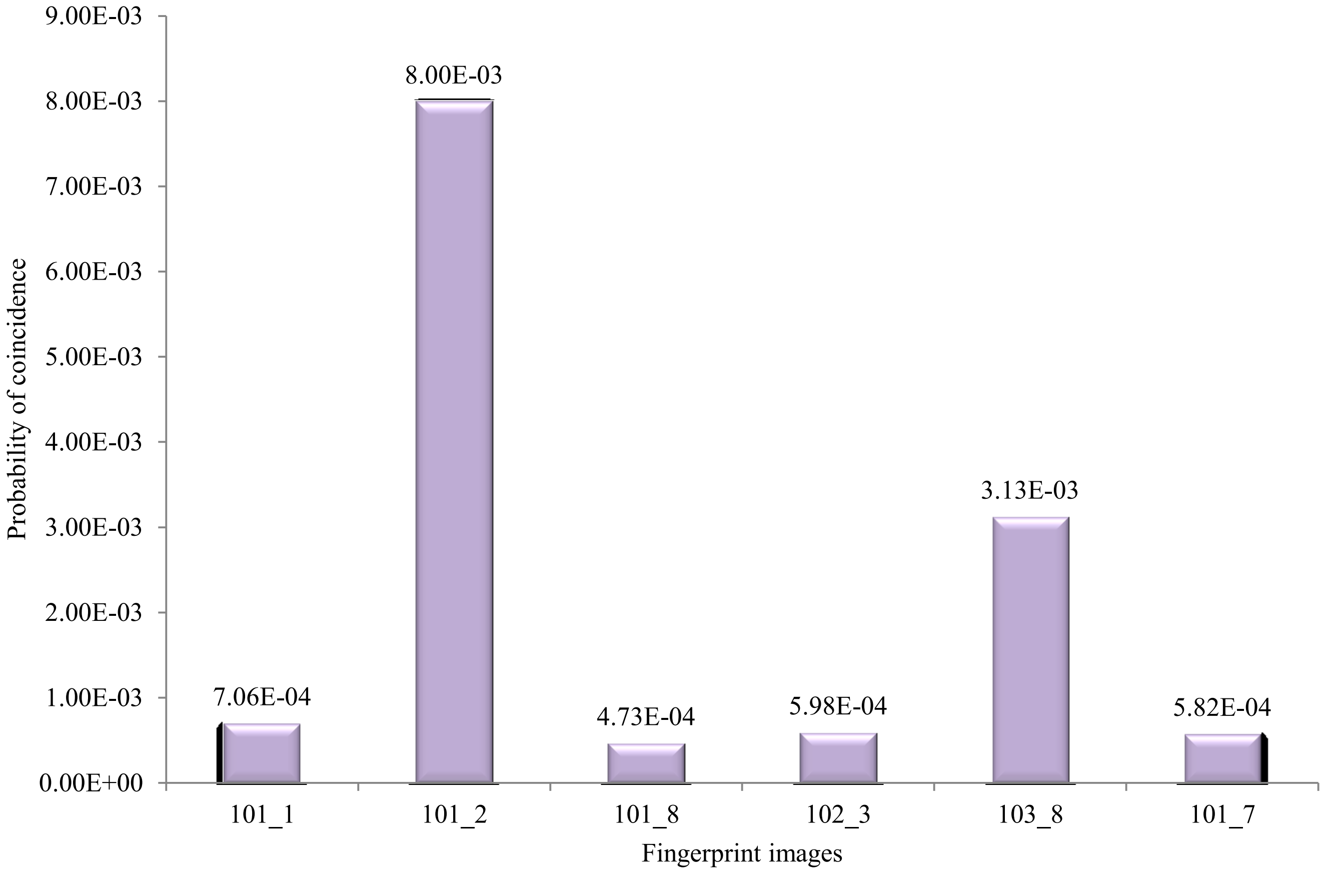
# Variation in number of minutiae points for different fingerprint images



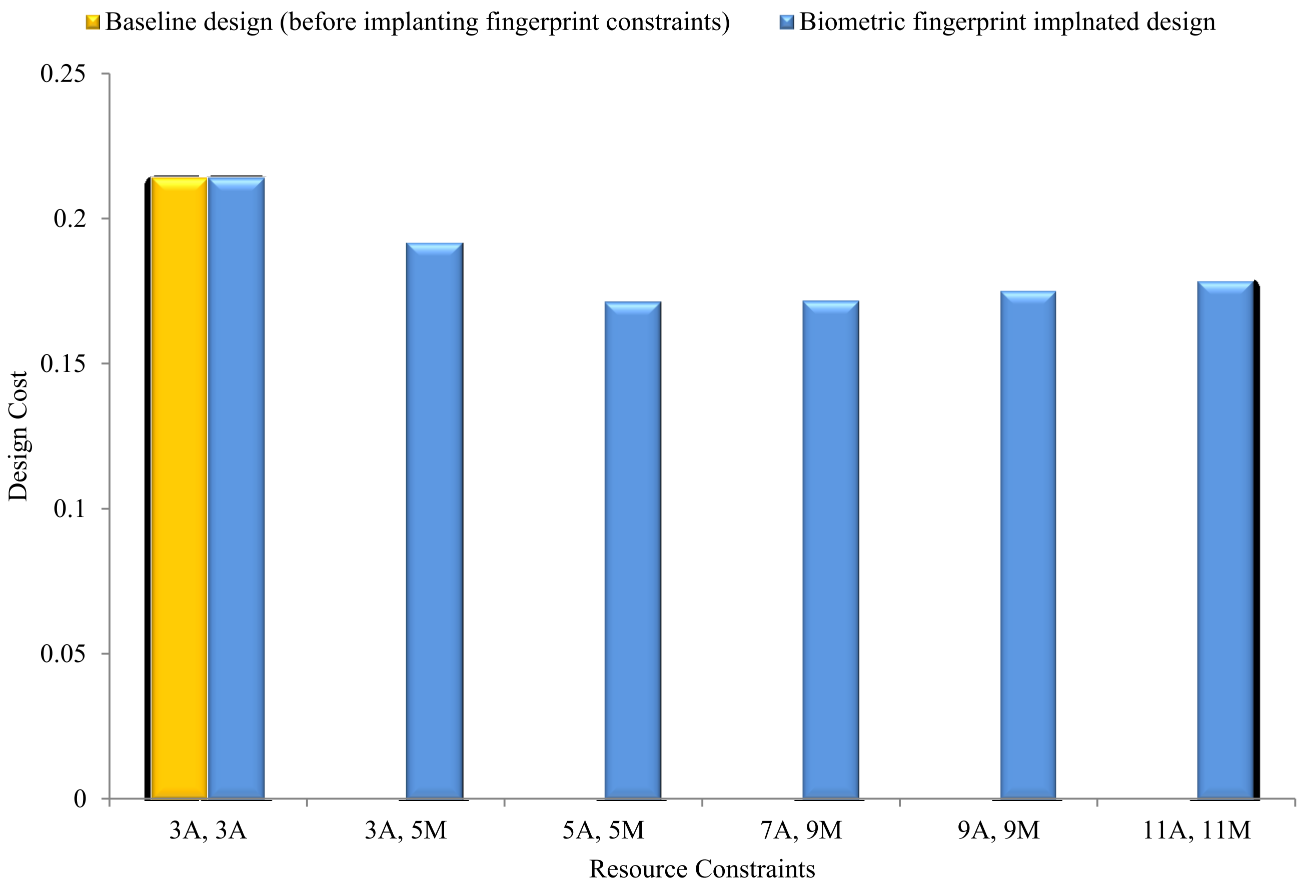
# Variation in total number of constraints (k1) for different fingerprint images



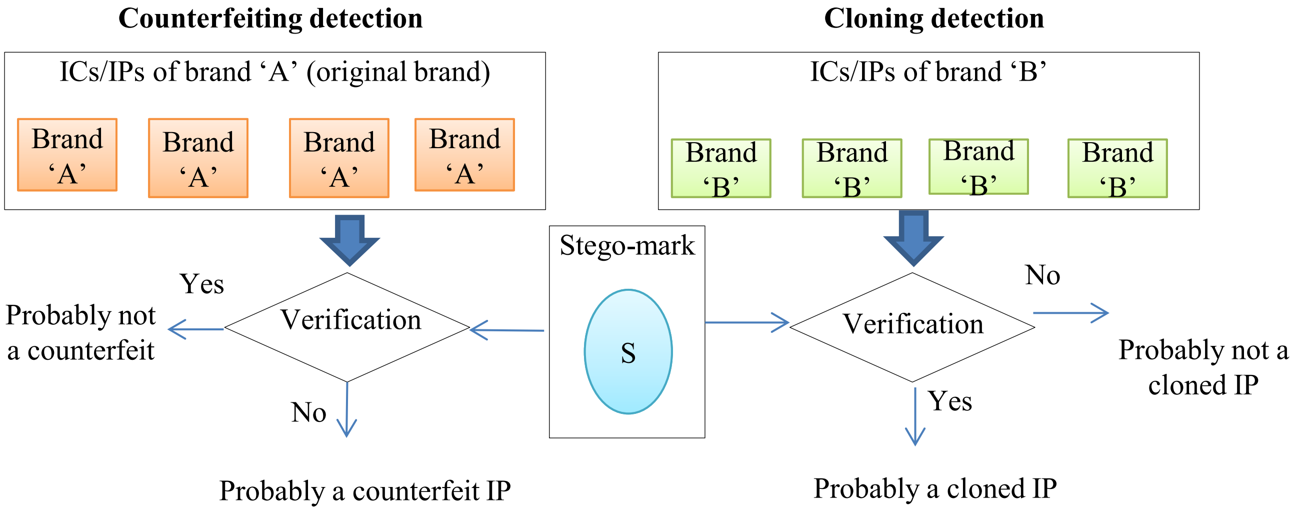
# Variation in probability of coincidence for different fingerprint images



# Comparison of design cost of JPEG compression hardware accelerator before and after implanting fingerprint constraints

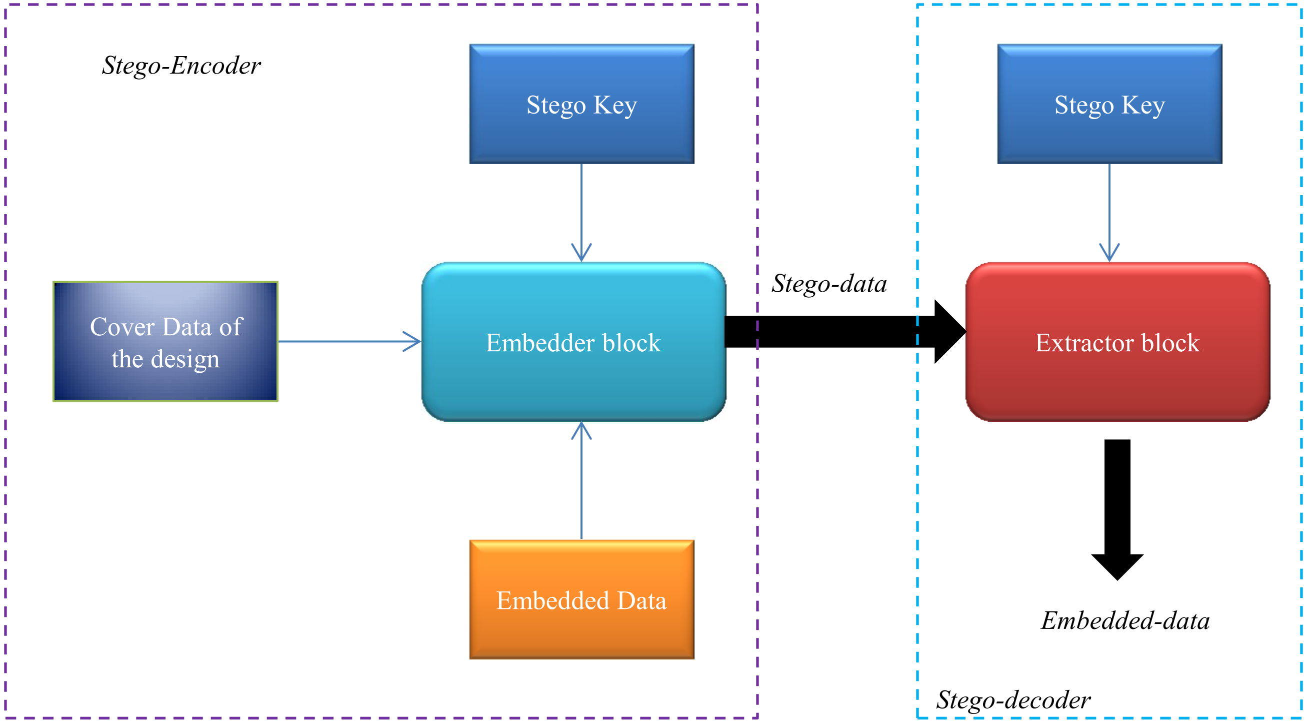


# Counterfeiting/cloning detection using proposed steganography



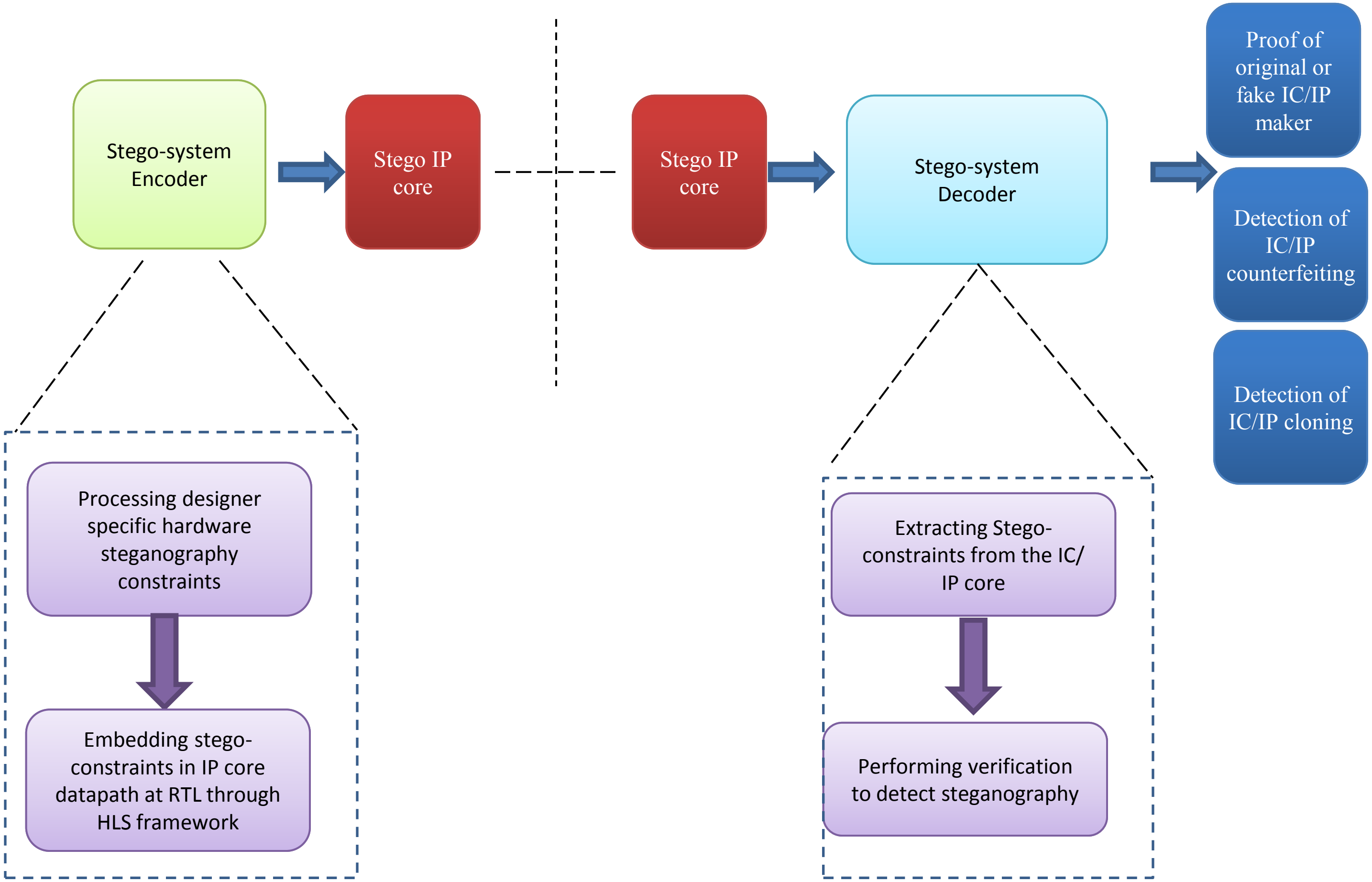
**Anirban Sengupta**, Mahendra Rathor "IP Core Steganography for Protecting DSP Kernels used in CE Systems", **IEEE Transactions on Consumer Electronics (TCE)** , Volume: 65 , Issue: 4 , Nov. 2019, pp. 506 - 515

# Basic Steganography Model



**Anirban Sengupta**, Mahendra Rathor "IP Core Steganography for Protecting DSP Kernels used in CE Systems", **IEEE Transactions on Consumer Electronics (TCE)** , Volume: 65 , Issue: 4 , Nov. 2019, pp. 506 - 515

# Hardware Steganography Encoding-Decoding Process



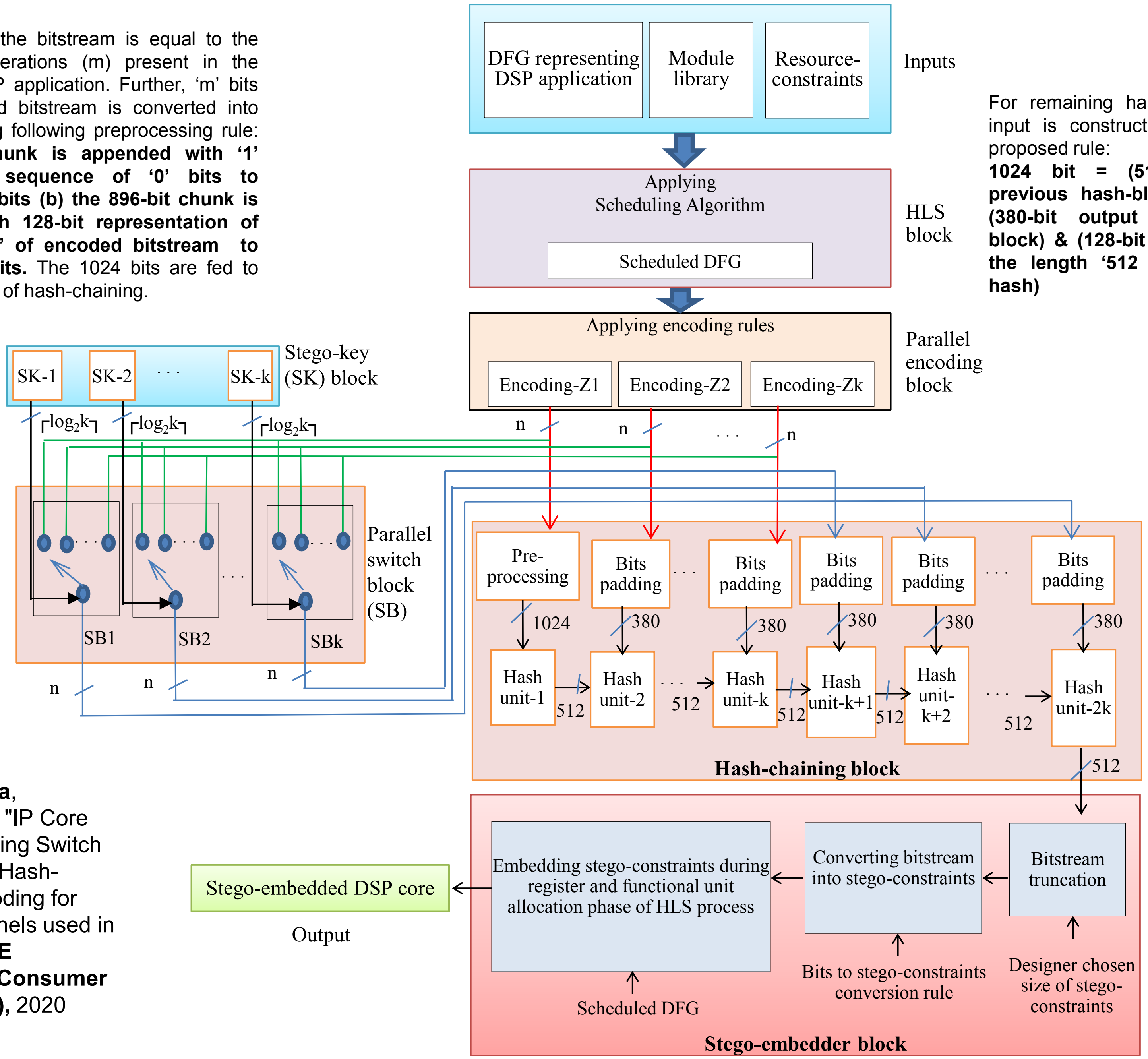
Stego-system Encoder
Stego IP core
Stego IP core
Stego-system Decoder
Proof of original or fake IC/IP maker
Detection of IC/IP counterfeiting
Detection of IC/IP cloning
Processing designer specific hardware steganography constraints
Embedding stego-constraints in IP core datapath at RTL through HLS framework
Extracting Stego-constraints from the IC/IP core
Performing verification to detect steganography

**Anirban Sengupta** "Frontiers in Securing IP Cores - Forensic detective control and obfuscation techniques", **The Institute of Engineering and Technology (IET),** 2020, ISBN-10: 1-83953-031-6, ISBN-13: 978-1-83953-031-9

**Anirban Sengupta**, Mahendra Rathor "IP Core Steganography for Protecting DSP Kernels used in CE Systems", **IEEE Transactions on Consumer Electronics (TCE)** , Volume: 65 , Issue: 4 , Nov. 2019, pp. 506 - 515

# Securing DSP cores using key-triggered hash-chaining based steganography

The length of the bitstream is equal to the number of operations (m) present in the respective DSP application. Further, 'm' bits of the encoded bitstream is converted into 1024 bits using following preprocessing rule: **(a) m-bit chunk is appended with '1' followed by sequence of '0' bits to generate 896 bits (b) the 896-bit chunk is appended with 128-bit representation of the length 'm' of encoded bitstream to obtain 1024 bits.** The 1024 bits are fed to first hash-block of hash-chaining.



For remaining hash-blocks, 1024-bit input is constructed using following proposed rule: **1024 bit = (512-bit output of previous hash-block) & "1000" & (380-bit output of bits-padding block) & (128-bit representation of the length '512 bits' of previous hash)**

**Anirban Sengupta**, Mahendra Rathor, "IP Core Steganography using Switch based Key-driven Hash-chaining and Encoding for Securing DSP kernels used in CE Systems", **IEEE Transactions on Consumer Electronics (TCE),** 2020

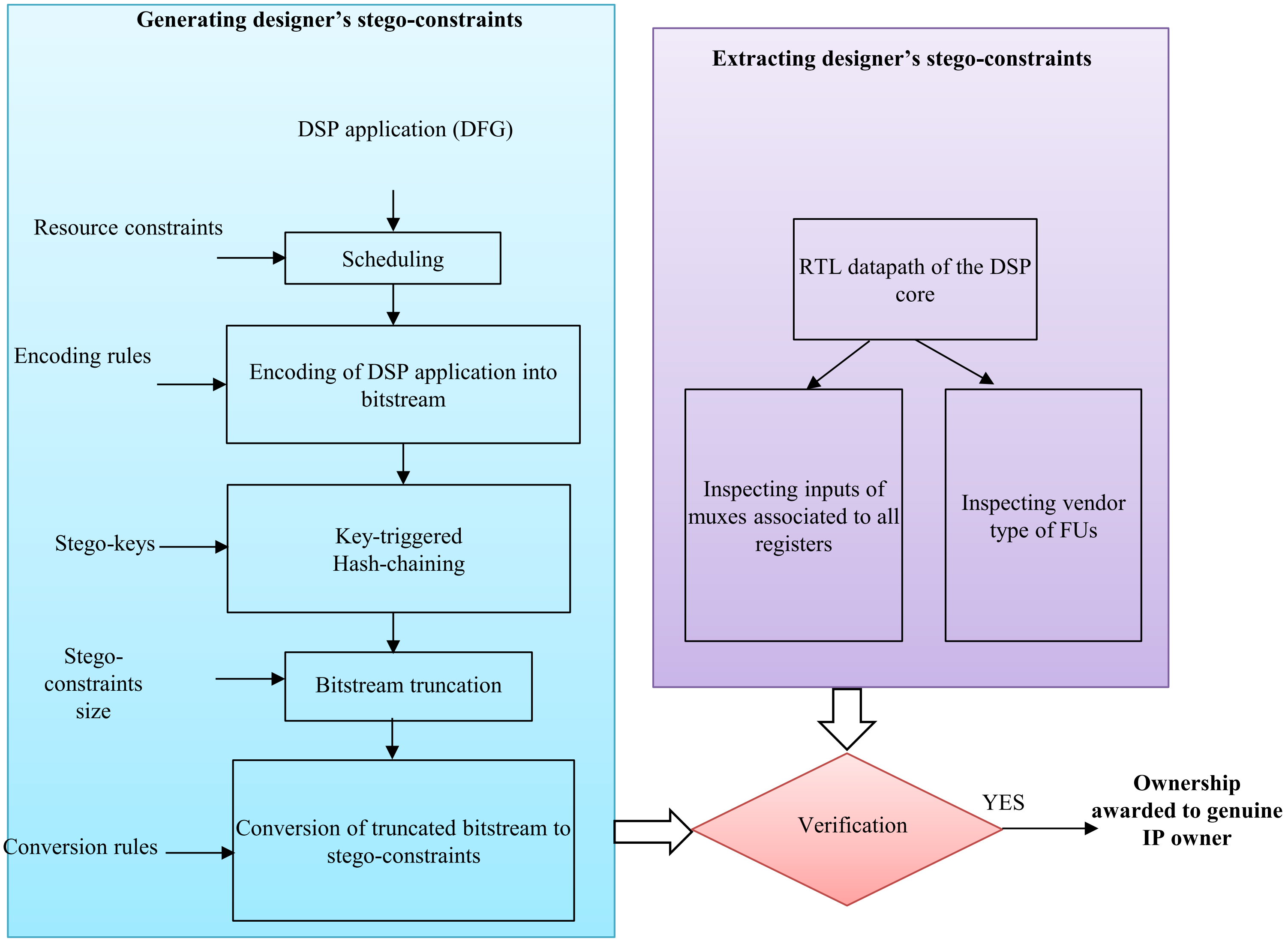
# Encoding in steganography

## Encoding Rules to Convert DSP Application into Bitstream Representations

| Proposed encoding | Encoding rule (Condition and encoded bit) | |
|---|---|---|
| E1 | If opn# and corresponding CS # are both even | 0 |
| | Otherwise | 1 |
| E2 | If opn# and corresponding CS# are of same parity (both even or both odd parity) | 0 |
| | If opn# and corresponding CS# are of different parity | 1 |
| E3 | If opn# and corresponding CS # are both odd | 0 |
| | Otherwise | 1 |
| E4 | If opn# and corresponding CS# are of different parity | 0 |
| | If opn# and corresponding CS# are of same parity | 1 |
| E5 | If opn# and corresponding CS# are both prime | 0 |
| | Otherwise | 1 |
| E6 | If opn# and corresponding CS# are both prime | 1 |
| | Otherwise | 0 |
| E7 | If GCD of opn# and corresponding CS# is 1 | 0 |
| | If GCD of opn# and corresponding CS# is not 1 | 1 |
| E8 | If (opn#) mod (corresponding CS#) is 0 | 0 |
| | If (opn#) mod (corresponding CS#) is not 0 | 1 |
| E9 | If CS# is equal to 2nd odd sequence of opn# | 0 |
| | Otherwise | 1 |

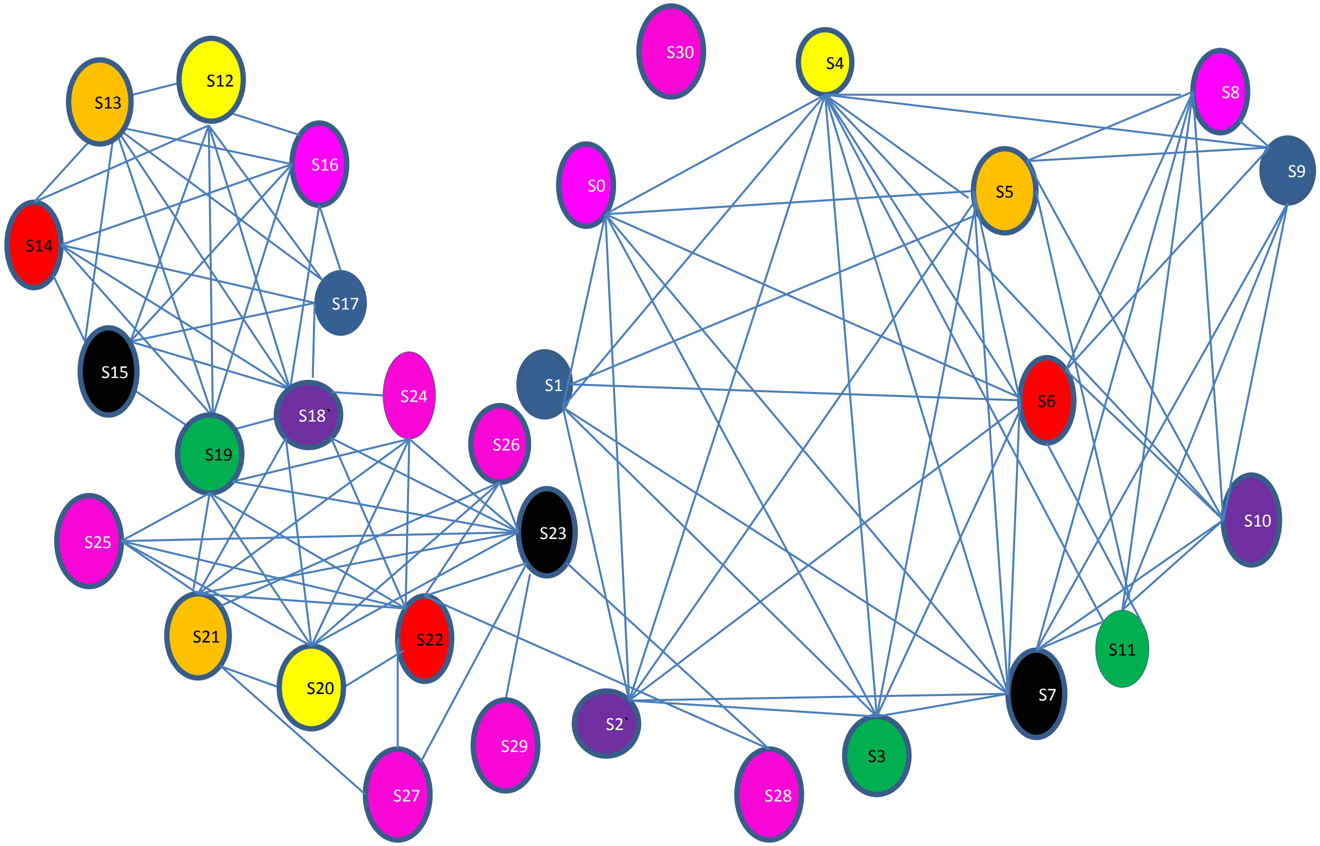
**Anirban Sengupta**, Mahendra Rathor, "IP Core Steganography using Switch based Key-driven Hash-chaining and Encoding for Securing DSP kernels used in CE Systems", **IEEE Transactions on Consumer Electronics (TCE),** 2020

# Detection process of steganography

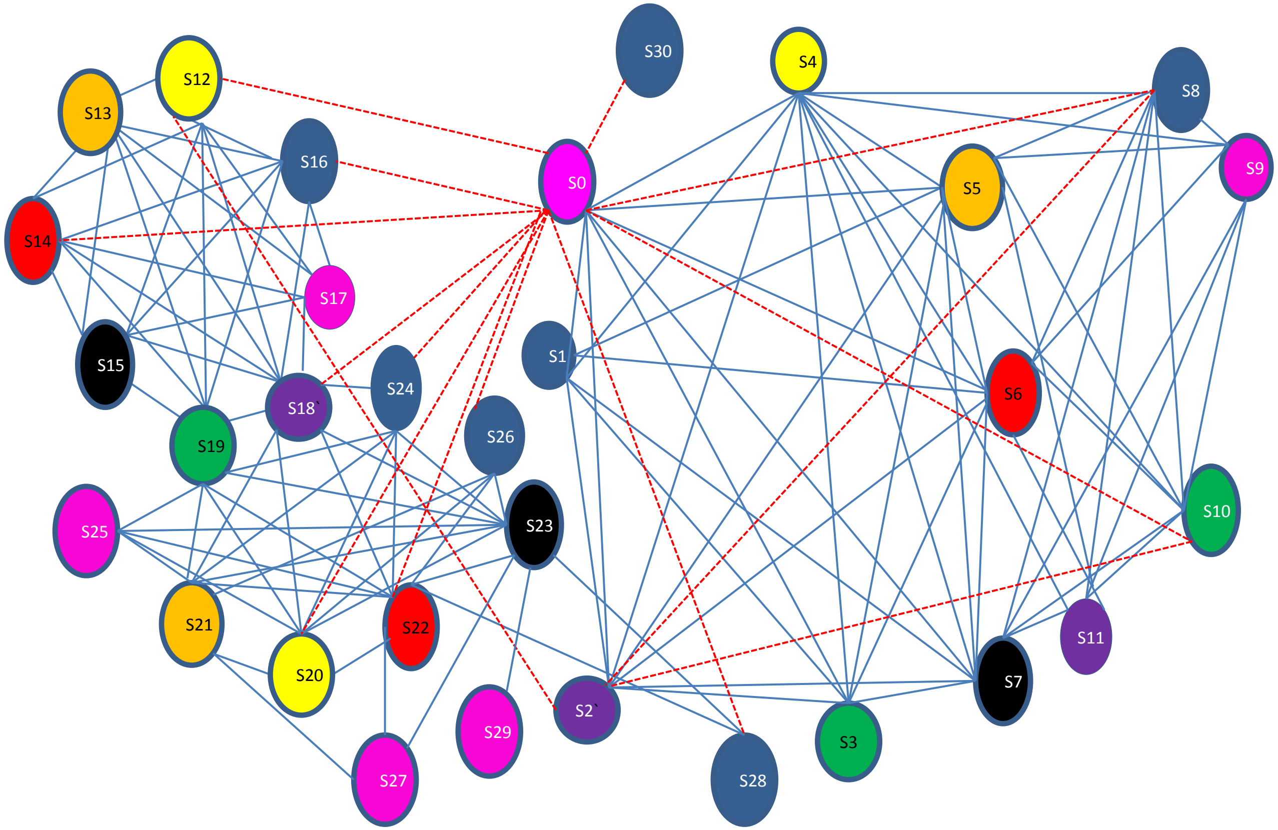

Generating designer's stego-constraints
DSP application (DFG)
Resource constraints
Scheduling
Encoding rules
Encoding of DSP application into bitstream
Stego-keys
Key-triggered Hash-chaining
Stego-constraints size
Bitstream truncation
Conversion rules
Conversion of truncated bitstream to stego-constraints
Extracting designer's stego-constraints
RTL datapath of the DSP core
Inspecting inputs of muxes associated to all registers
Inspecting vendor type of FUs
Verification
YES
Ownership awarded to genuine IP owner

**Anirban Sengupta**, Mahendra Rathor, "IP Core Steganography using Switch based Key-driven Hash-chaining and Encoding for Securing DSP kernels used in CE Systems", **IEEE Transactions on Consumer Electronics (TCE),** 2020

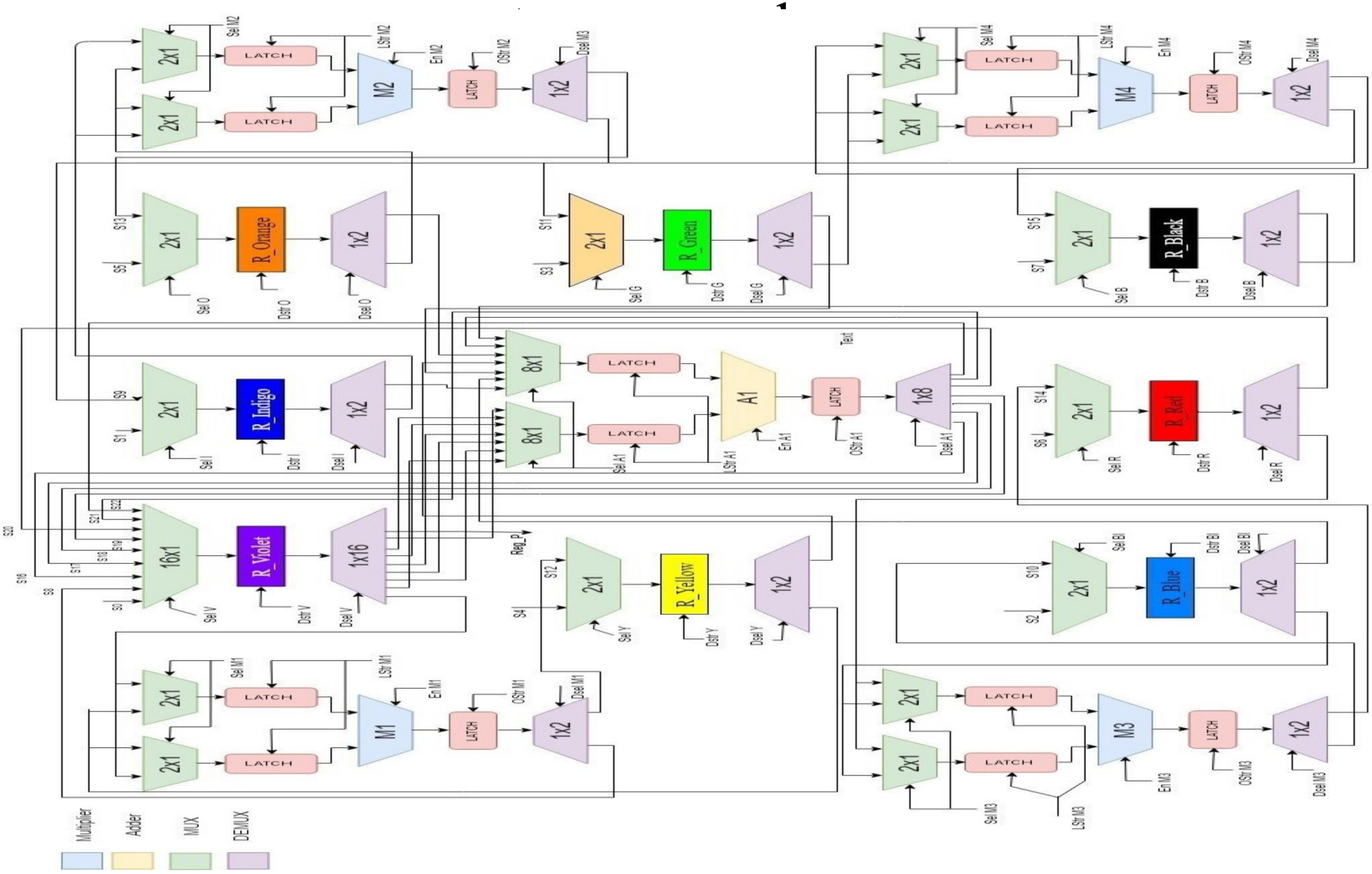
# CIG of FIR filter hardware accelerator (IP core) before steganography



# CIG of FIR filter hardware accelerator (IP core) after steganography

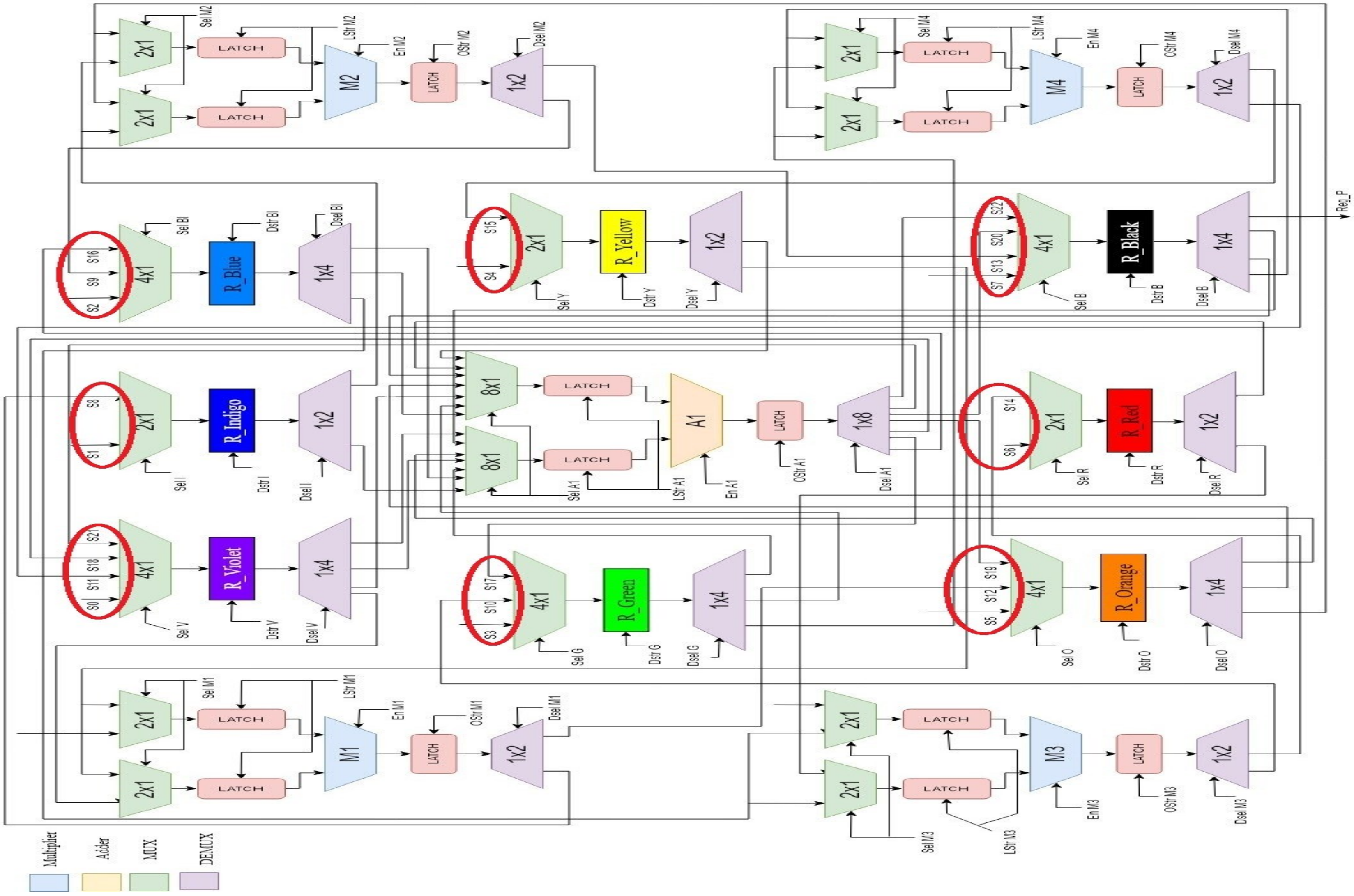


# RTL datapath of 8-point DCT before



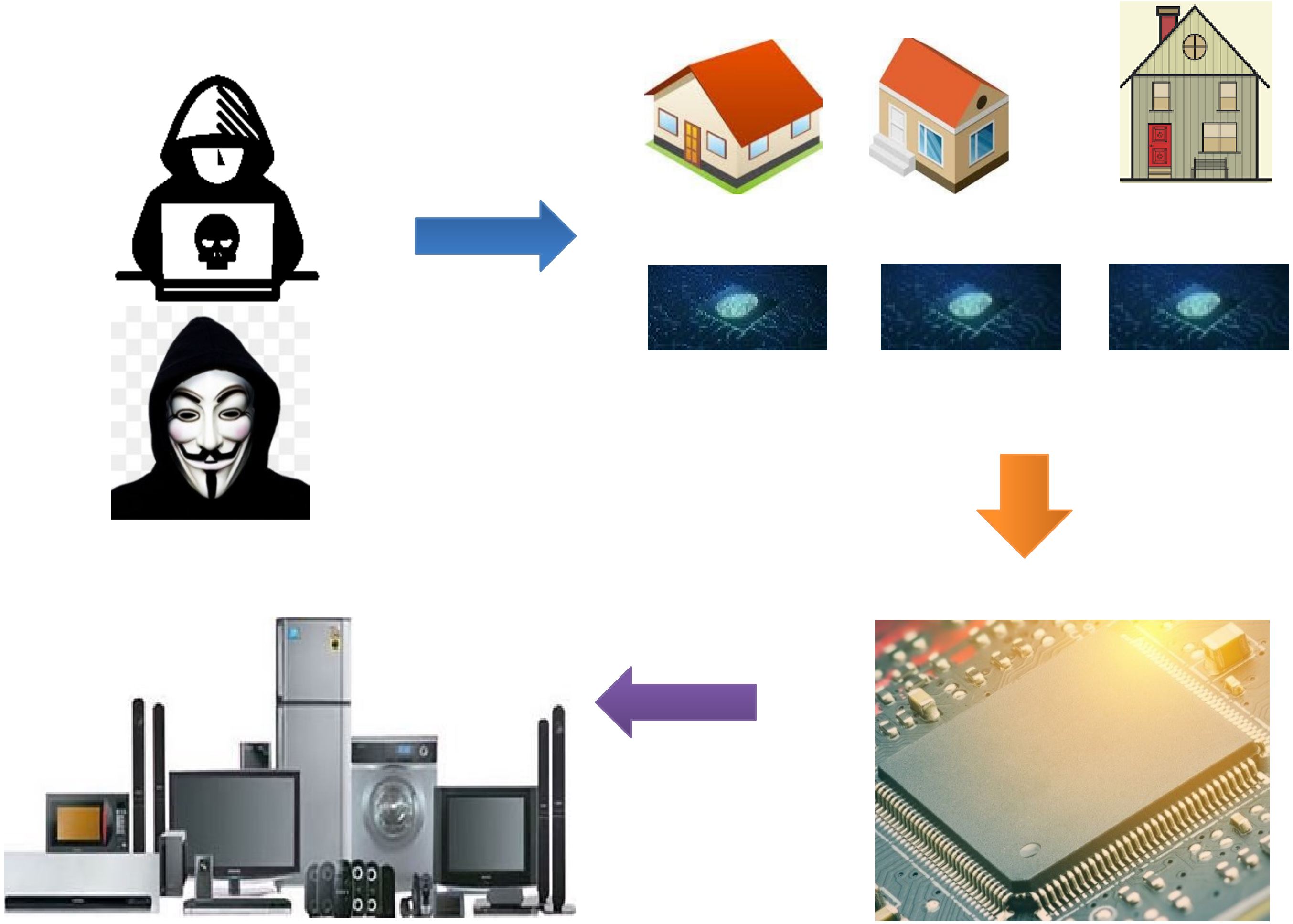
**Anirban Sengupta**, Mahendra Rathor "IP Core Steganography for Protecting DSP Kernels used in CE Systems", **IEEE Transactions on Consumer Electronics (TCE)** , Volume: 65 , Issue: 4 , Nov. 2019, pp. 506 - 515

# RTL datapath of 8-point DCT **after** Steganography



**Anirban Sengupta**, Mahendra Rathor "IP Core Steganography for Protecting DSP Kernels used in CE Systems", **IEEE Transactions on Consumer Electronics (TCE)** , Volume: 65 , Issue: 4 , Nov. 2019, pp. 506 - 515

# IP core protection using Digital Signature



Consumer Electronic devices

SOC with third party IPs

**Anirban Sengupta**, E. Ranjith Kumar, N. Prajwal Chandra "Embedding Digital Signature using Encrypted-Hashing for Protection of DSP cores in CE", **IEEE Transactions on Consumer Electronics (TCE)**, Volume: 65, Issue:3, Aug 2019, pp. 398 - 407

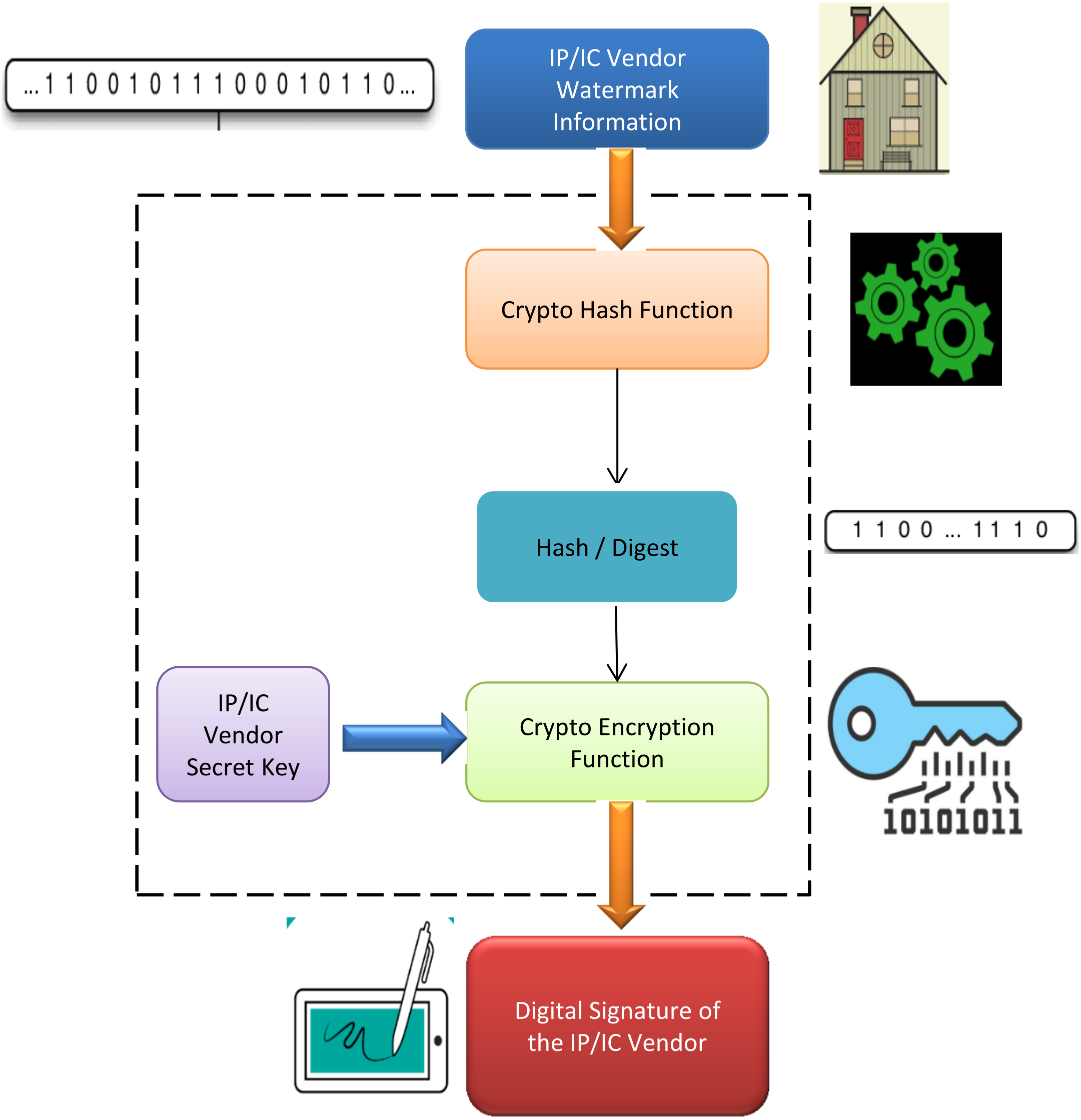
# IP core protection using Digital Signature

- **A novel crypto digital signature approach is presented which incorporates following security modules**
  - Crypto hash function- SHA-512
  - Crypto encryption function- RSA
  - Encoding

- **The generic steps of generating digital signature:**
  - Generate a Bit-stream representation of DSP Core.
  - Performing SHA-512
  - Post-processing Step1
  - RSA Encryption
  - Post-processing Step 2

**Anirban Sengupta**, E. Ranjith Kumar, N. Prajwal Chandra "Embedding Digital Signature using Encrypted-Hashing for Protection of DSP cores in CE", **IEEE Transactions on Consumer Electronics (TCE)**, Volume: 65, Issue:3, Aug 2019, pp. 398 - 407

# High-level process of creating digital signature for IP cores



…1 1 0 0 1 0 1 1 1 0 0 0 1 0 1 1 0…

IP/IC Vendor Watermark Information

Crypto Hash Function

Hash / Digest

1 1 0 0 … 11 1 0

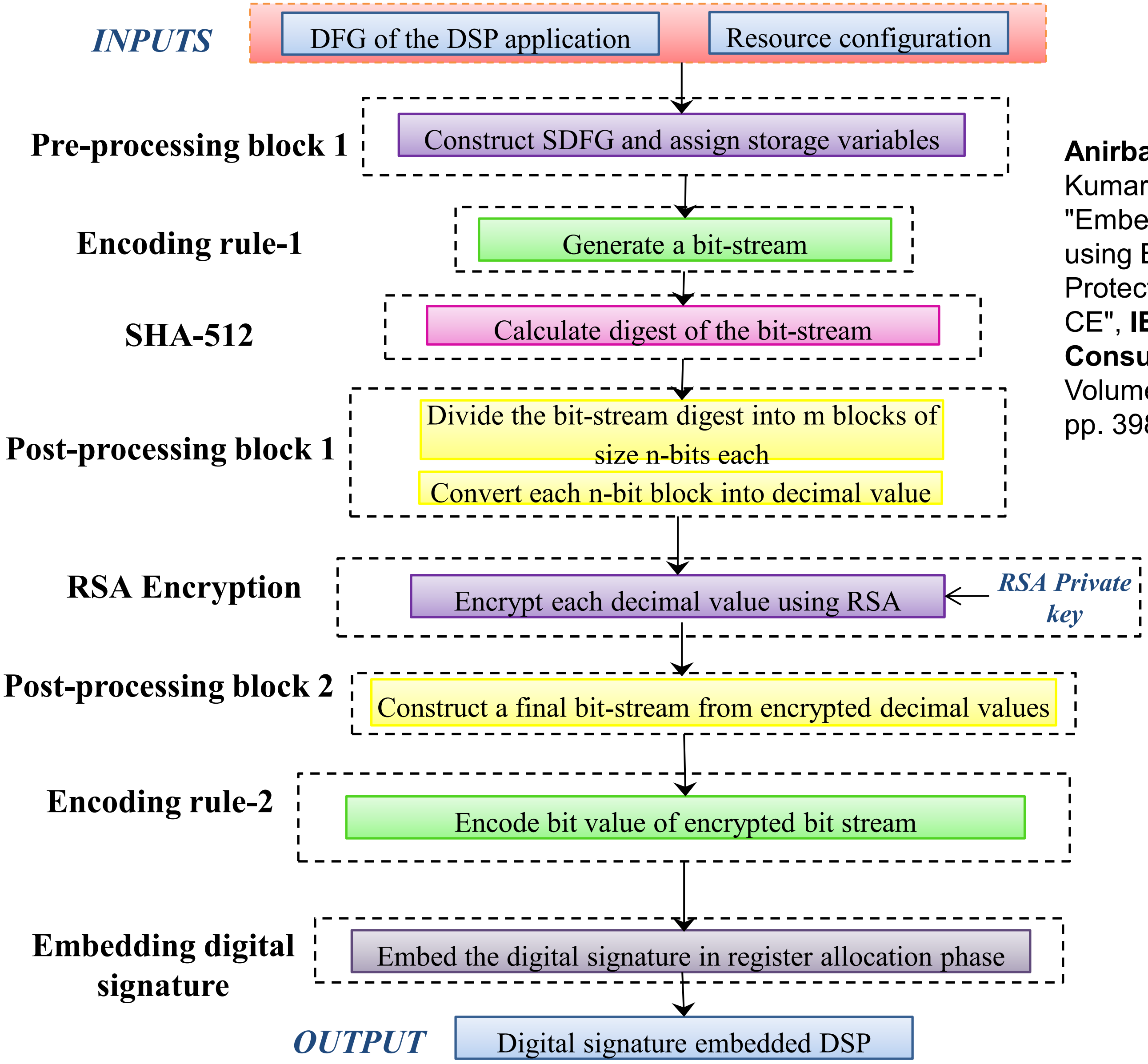
IP/IC Vendor Secret Key

Crypto Encryption Function

Digital Signature of the IP/IC Vendor

**Anirban Sengupta**, E. Ranjith Kumar, N. Prajwal Chandra "Embedding Digital Signature using Encrypted-Hashing for Protection of DSP cores in CE", **IEEE Transactions on Consumer Electronics (TCE)**, Volume: 65, Issue:3, Aug 2019, pp. 398 - 407

# Flow of the approach


INPUTS
DFG of the DSP application
Resource configuration
Pre-processing block 1
Construct SDFG and assign storage variables
Encoding rule-1
Generate a bit-stream
SHA-512
Calculate digest of the bit-stream
Post-processing block 1
Divide the bit-stream digest into m blocks of size n-bits each
Convert each n-bit block into decimal value
RSA Encryption
Encrypt each decimal value using RSA
RSA Private key
Post-processing block 2
Construct a final bit-stream from encrypted decimal values
Encoding rule-2
Encode bit value of encrypted bit stream
Embedding digital signature
Embed the digital signature in register allocation phase
OUTPUT
Digital signature embedded DSP

**Anirban Sengupta**, E. Ranjith Kumar, N. Prajwal Chandra "Embedding Digital Signature using Encrypted-Hashing for Protection of DSP cores in CE", **IEEE Transactions on Consumer Electronics (TCE)**, Volume: 65, Issue:3, Aug 2019, pp. 398 - 407

# Performing SHA-512 and Post Processing-1

## ➢Performing SHA-512

- Generates decimal digest of DSP core
- The collision resistance and deterministic properties of SHA-512 ensures that the generated hash digest carrying vendor secret mark is unique for an IP core design.

## ➢ Post-Processing-1

- Divide the bitstream digest into m blocks of size n bits each
- Convert each n-bit block into decimal value

**Anirban Sengupta**, E. Ranjith Kumar, N. Prajwal Chandra "Embedding Digital Signature using Encrypted-Hashing for Protection of DSP cores in CE", **IEEE Transactions on Consumer Electronics (TCE)**, Volume: 65, Issue:3, Aug 2019, pp. 398 - 407

# RSA Encryption

- RSA is an asymmetric key encryption algorithm in which two distinct keys (private key and public key) are involved in the cryptography.
- It is used to sign the hash digest of vendor secret mark information to ensure authentication of the genuine owner.

**Inputs (128 bits) and outputs of RSA module**

| RSA Decimal Input | Encrypted Decimal Output | Encrypted Binary Equivalent |
|---|---|---|
| **328629211327023509667307560016780722176** | 259269232367594955022606516388222677830 | 110000110000110............011001101000110 |
| **338967726051337675217876235804913696768** | 103304422214721939828321919365106451481 | 100110110110111............100010000011001 |
| **74508071724950413296959519603599240546** | 24682247863022431965512042630223003075 | 100101001000110............111110111000011 |
| **14047629289186758734138225185020455483** | 289411796889479622387067677582110256178 | 110110011011101............110110000110010 |

**Anirban Sengupta**, E. Ranjith Kumar, N. Prajwal Chandra "Embedding Digital Signature using Encrypted-Hashing for Protection of DSP cores in CE", **IEEE Transactions on Consumer Electronics (TCE)**, Volume: 65, Issue:3, Aug 2019, pp. 398 - 407

# Post-processing Block 2

- The encrypted decimal values— output of RSA module— are provided as input to the post-processing block 2.
- Each decimal value is converted to binary and these individual binary streams are concatenated to form a single bit-stream.
- This encrypted-hashed bit-stream is referred to as **Digital Signature**. The digital signature size can be selected based on vendor's choice from the continuous bit-stream.
- For instance, if the vendor selects digital signature size as 15, then the first 15 bits of the bit-stream is the digital signature.

**Anirban Sengupta**, E. Ranjith Kumar, N. Prajwal Chandra "Embedding Digital Signature using Encrypted-Hashing for Protection of DSP cores in CE", **IEEE Transactions on Consumer Electronics (TCE)**, Volume: 65, Issue:3, Aug 2019, pp. 398 - 407

# Embedding Digital Signature

Having created the digital signature, the next step is to embed it in the design. The steps to implant the digital signature are stated below:

- **Mapping the digital signature bits to watermarking constraints**
  - Using the following encoding rule:
    - If bit = ‘0’, then additional edge is added between node pair (prime, prime) in a colored interval graph.
    - If bit = ‘1’, then additional edge is added between node pair (even, even) in a colored interval graph.
- **Embedding the watermarking constraints**.
  - Watermark constraints are embedded in register allocation step during HLS (And it is performed through colored interval graph framework).
  - These hidden constraints act as additional constraints to be imposed besides the regular design constraints of the design

**Anirban Sengupta**, E. Ranjith Kumar, N. Prajwal Chandra "Embedding Digital Signature using Encrypted-Hashing for Protection of DSP cores in CE", **IEEE Transactions on Consumer Electronics (TCE)**, Volume: 65, Issue:3, Aug 2019, pp. 398 - 407

# Bit stream Generation

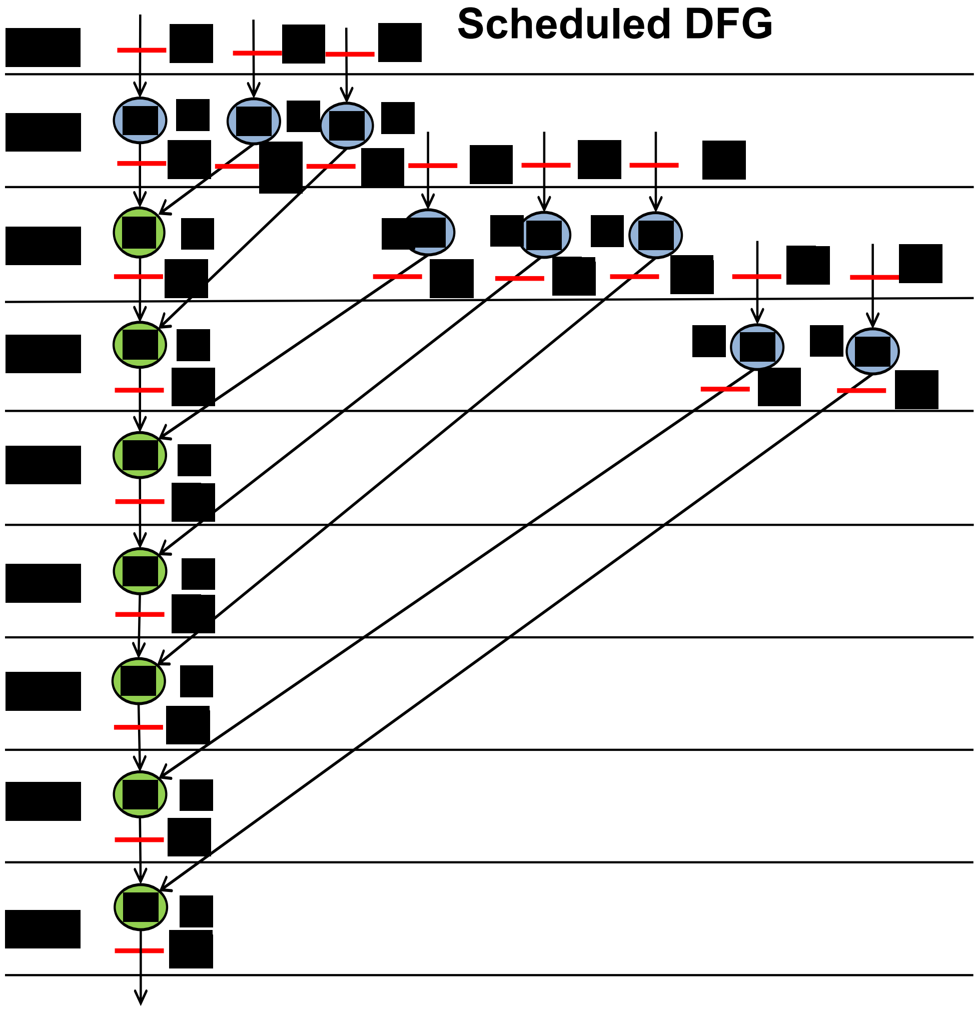
- Based on encoding rule-1

| Operation number (OPN) | Corresponding control step (CS) number | Encoded bit |
|---|---|---|
| **Even** | Even | 0 |
| **Odd** | Even | 1 |
| **Even** | Odd | 1 |
| **Odd** | Odd | 0 |

**Anirban Sengupta**, E. Ranjith Kumar, N. Prajwal Chandra "Embedding Digital Signature using Encrypted-Hashing for Protection of DSP cores in CE", **IEEE Transactions on Consumer Electronics (TCE)**, Volume: 65, Issue:3, Aug 2019, pp. 398 - 407

# Generating the Bit-stream of DCT Core

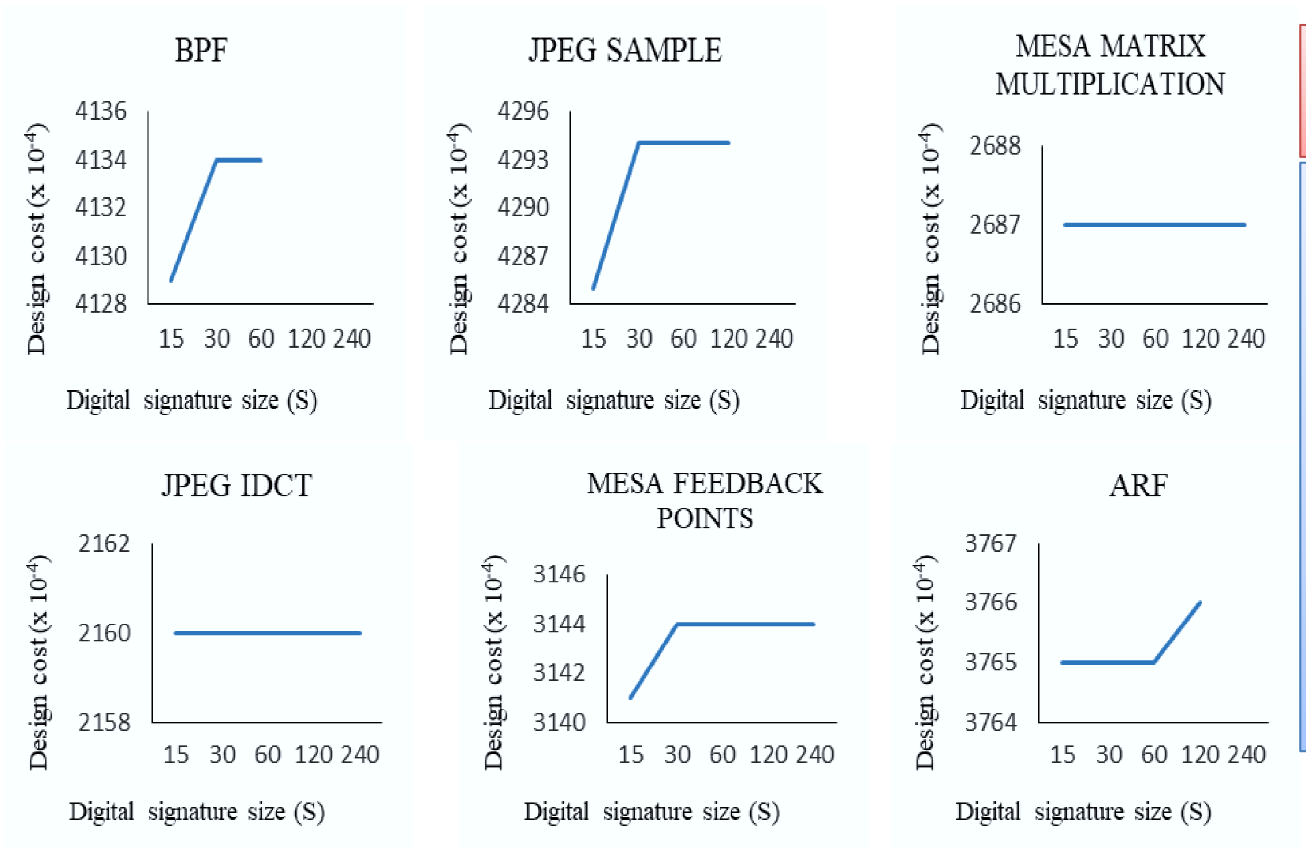
| Operation Number | Control Step Number | Bit generated |
|---|---|---|
| 1 | 1 | 0 |
| 2 | 1 | 1 |
| 3 | 2 | 1 |
| 4 | 1 | 1 |
| 5 | 3 | 0 |
| 6 | 2 | 0 |
| 7 | 4 | 1 |
| 8 | 2 | 0 |
| 9 | 5 | 0 |
| 10 | 2 | 0 |
| 11 | 6 | 1 |
| 12 | 3 | 1 |
| 13 | 7 | 0 |
| 14 | 3 | 1 |
| 15 | 8 | 1 |



**Anirban Sengupta**, E. Ranjith Kumar, N. Prajwal Chandra "Embedding Digital Signature using Encrypted-Hashing for Protection of DSP cores in CE", **IEEE Transactions on Consumer Electronics (TCE)**, Volume: 65, Issue:3, Aug 2019, pp. 398 - 407

# Experimental Results

- **Graphical Representation of Design Cost for different benchmarks**



**Design cost**

$$C_f(X_i) = \phi_1 \frac{L_T}{L_{max}} + \phi_2 \frac{A_T}{A_{max}}$$

$L_T$ =design latency
$A_T$ =hardware area
$L_{max}$= maximum execution latency
$A_{max}$ = maximum hardware area.

$\phi_1$, $\phi_2$ represent the user specified weights both fixed at 0.5 to assign equal preference

**Anirban Sengupta**, E. Ranjith Kumar, N. Prajwal Chandra "Embedding Digital Signature using Encrypted-Hashing for Protection of DSP cores in CE", **IEEE Transactions on Consumer Electronics (TCE)**, Volume: 65, Issue:3, Aug 2019, pp. 398 - 407

# Experimental Results

➢ **Evaluation of Robustness Using Probability of Coincidence (Pc)**

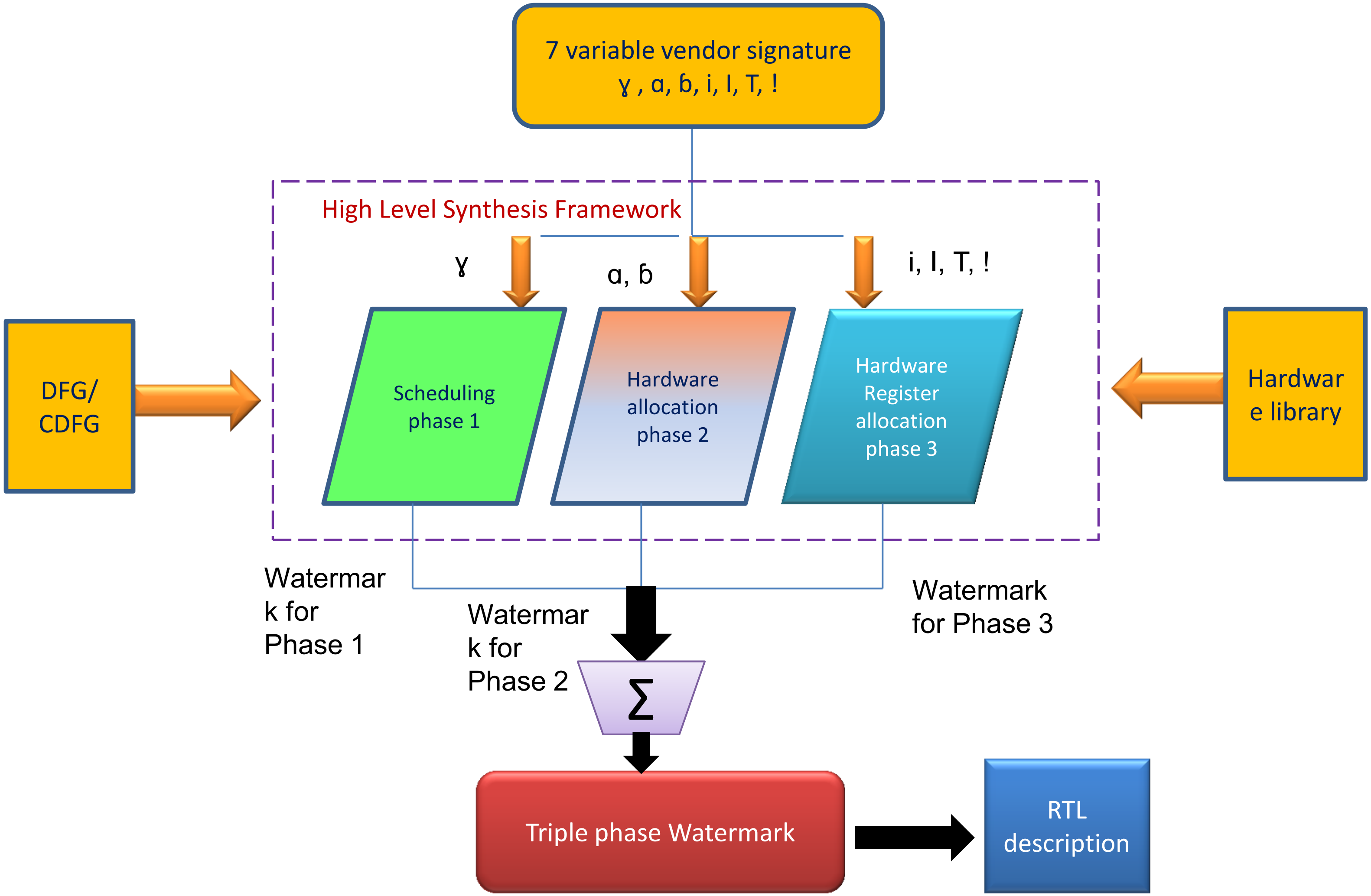
$$P_c = \left(1 - \frac{1}{c}\right)^S$$

‘c’ denotes the number of colours used in the CIG and
‘S’ denotes the digital signature size

| Benchmarks | c | Size of Digital signature (S) | | | | |
|---|---|---|---|---|---|---|
| | | S = 15 | S = 30 | S = 60 | S = 120 | S = 240 |
| | | $P_c$ | $P_c$ | $P_c$ | $P_c$ | $P_c$ |
| BPF | 6 | 0.0649 | $4.2127\text{x}10^{-3}$ | $1.7747\text{x}10^{-5}$ | $3.1496\text{x}10^{-10}$ | $9.9198\text{x}10^{-20}$ |
| JPEG SAMPLE | 10 | 0.2059 | 0.0424 | $1.7970\text{x}10^{-3}$ | $3.2292\text{x}10^{-6}$ | $1.0428\text{x}10^{-11}$ |
| JPEG IDCT | 29 | 0.5907 | 0.3490 | 0.1218 | 0.0148 | $2.1999\text{x}10^{-4}$ |
| MESA FEEDBACK POINTS | 17 | 0.4028 | 0.1622 | 0.0263 | $6.9267\text{x}10^{-4}$ | $4.7979\text{x}10^{-7}$ |
| ARF | 8 | 0.1349 | 0.0182 | $3.3150\text{x}10^{-4}$ | $1.0989\text{x}10^{-7}$ | $1.2076\text{x}10^{-14}$ |
| MESA MATRIX MULTIPLICATION | 23 | 0.5134 | 0.2635 | 0.0695 | $4.8237\text{x}10^{-3}$ | $2.3268\text{x}10^{-5}$ |

**Anirban Sengupta**, E. Ranjith Kumar, N. Prajwal Chandra "Embedding Digital Signature using Encrypted-Hashing for Protection of DSP cores in CE", **IEEE Transactions on Consumer Electronics (TCE)**, Volume: 65, Issue:3, Aug 2019, pp. 398 - 407

# High level overview of triple phase watermarking



7 variable vendor signature
ɣ , ɑ, ɓ, i, I, T, !

High Level Synthesis Framework

ɣ

ɑ, ɓ

i, I, T, !

DFG/
CDFG

Scheduling
phase 1

Hardware
allocation
phase 2

Hardware
Register
allocation
phase 3

Hardwar
e library

Watermar
k for
Phase 1

Watermar
k for
Phase 2

Watermark
for Phase 3

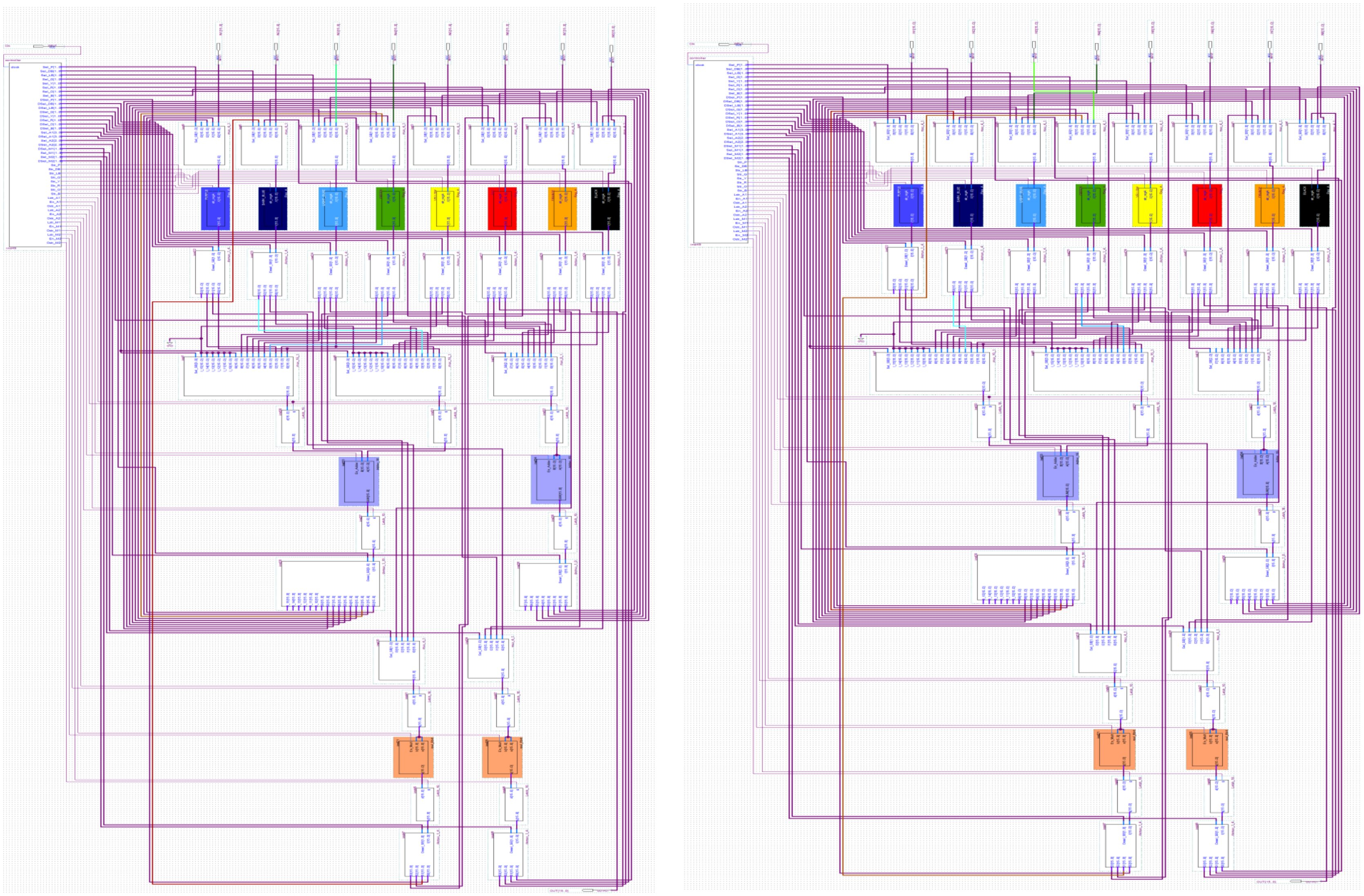
∑

Triple phase Watermark

RTL
description

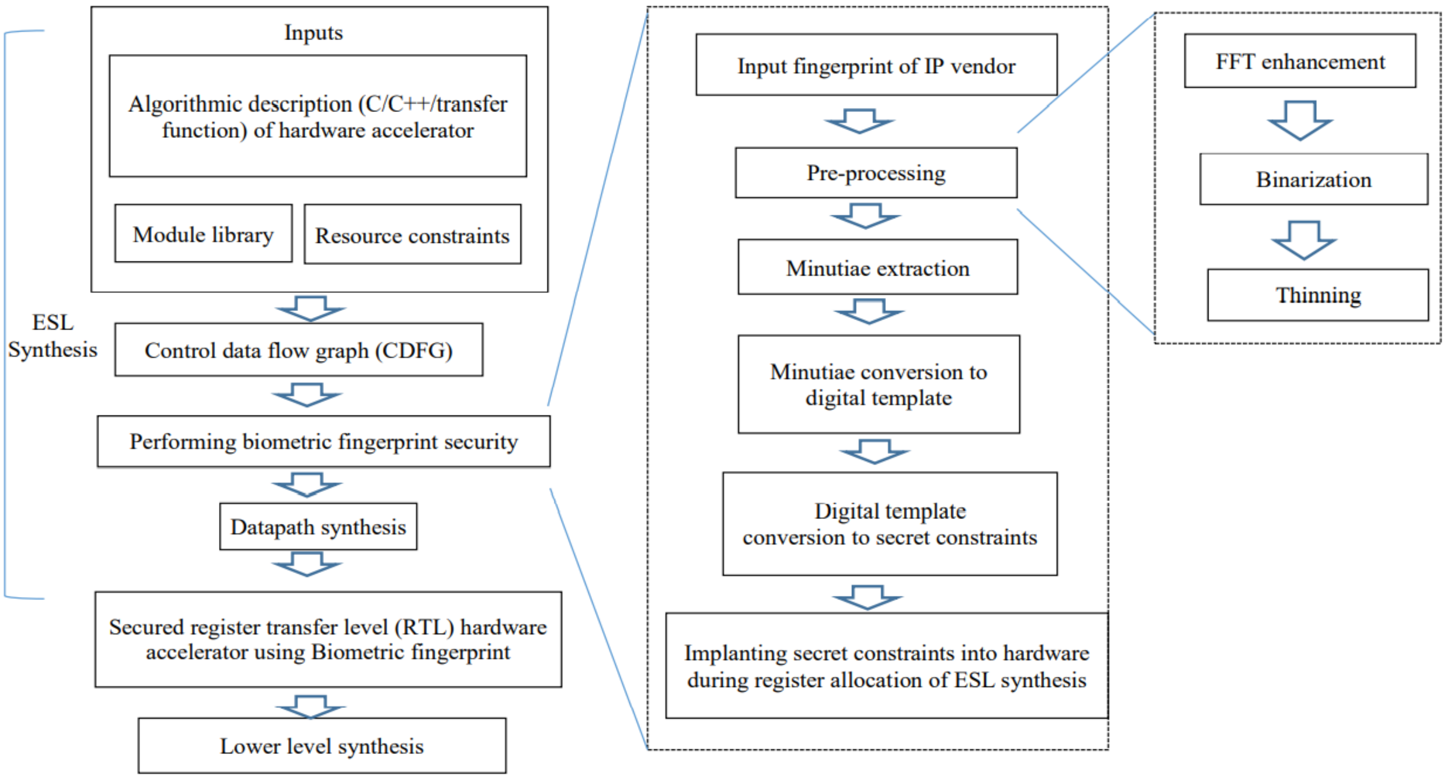
**Anirban Sengupta,** Dipanjan Roy, Saraju P Mohanty, "Triple-Phase Watermarking for Reusable IP Core Protection during Architecture Synthesis", **IEEE Transactions on Computer Aided Design of Integrated Circuits & Systems (TCAD),** Volume: 37, Issue: 4, April 2018, pp. 742 - 755

# Watermarked FIR Vs Non-Watermarked FIR at RTL



Anirban Sengupta, Dipanjan Roy, Saraju P Mohanty, "Triple-Phase Watermarking for Reusable IP Core Protection during Architecture Synthesis", **IEEE Transactions on Computer Aided Design of Integrated Circuits & Systems (TCAD),** Volume: 37, Issue: 4, April 2018, pp. 742 - 755

# Biometric Fingerprint for Hardware Security

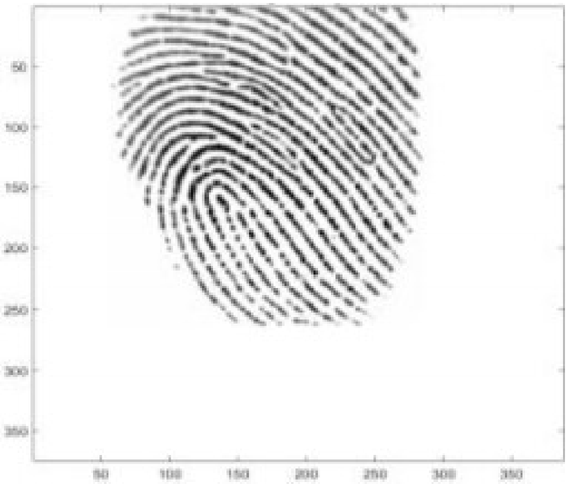

Inputs
Algorithmic description (C/C++/transfer function) of hardware accelerator
Module library
Resource constraints
ESL Synthesis
Control data flow graph (CDFG)
Performing biometric fingerprint security
Datapath synthesis
Secured register transfer level (RTL) hardware accelerator using Biometric fingerprint
Lower level synthesis
Input fingerprint of IP vendor
Pre-processing
Minutiae extraction
Minutiae conversion to digital template
Digital template conversion to secret constraints
Implanting secret constraints into hardware during register allocation of ESL synthesis
FFT enhancement
Binarization
Thinning

*(i)* *FFT enhancement*: The use of FFT on sets of pixels of the fingerprint image allows reconnection of broken ridges; finely separates the parallel ridges and also makes the ridges thick.

*(ii)* *Binarization*: The image with only two intensity values is called as binary image. This image usually shows only black or white, where black is represented by 0 and white is represented by 255. The elementary principle of binarization is to compare the pixel intensities with the threshold, and setting the pixels whose intensities are less than threshold, to 0 and the other to 255.

*(iii)* *Thinning*: In this process, the thickness of ridge lines is reduced to one pixel width by deleting pixels at the edge of ridge lines.

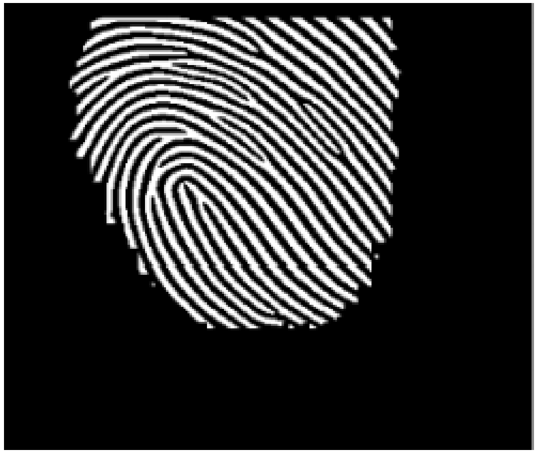
# Processing the Biometric Fingerprint



Original Image

Binarization

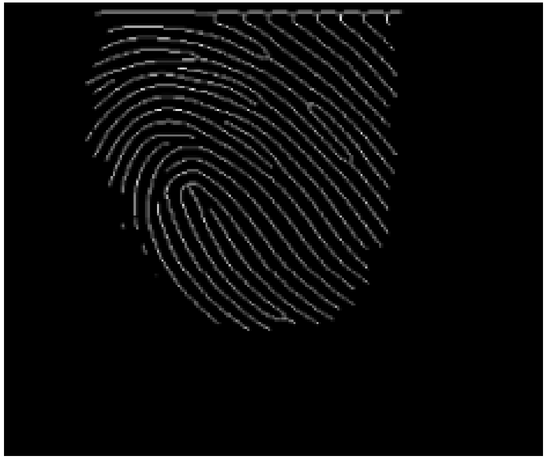
Setting pixel intensity to 0 or 1



Binary Image

Thinning

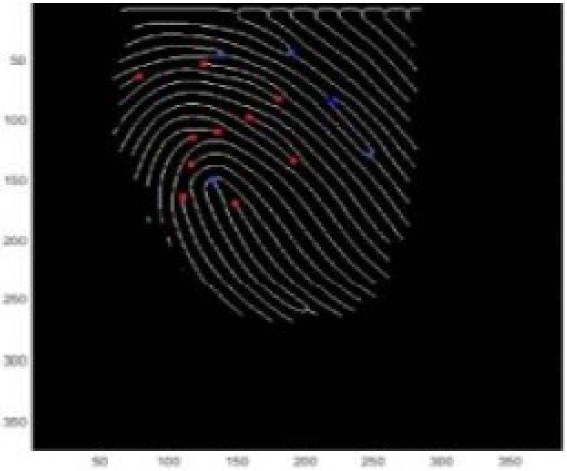
Reducing ridge line thickness to 1 pixel



Thinned Image

Minutiae point extraction

Applying CN algorithm to obtain ridge endings and bifurcations



Minutiae Points

# Extracting Minutiae Points

Surrounding pixels

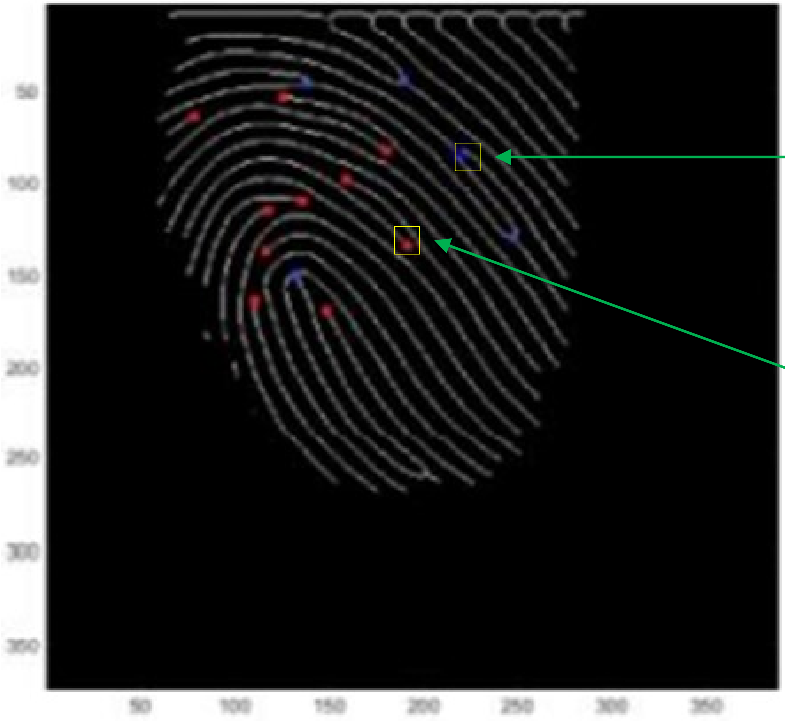
|  |  |  |
|---|---|---|
| $P_4$ | $P_3$ | $P_2$ |
| $P_5$ | $P$ | $P_1$ |
| $P_6$ | $P_7$ | $P_8$ |

Pixel of interest

Crossing Number (CN) algorithm

$$CN = 0.5 * \sum_{i=1}^{8} |P_i - P_{i+1}|$$

$P_i$ is the pixel value of the surrounding pixel.
$P_i = 0$ or $1$ and $P_1 = P_9$

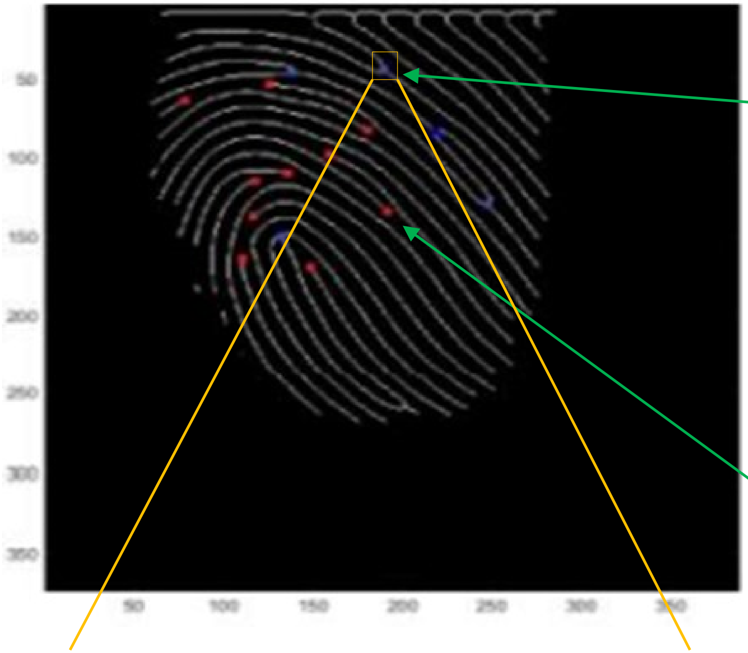


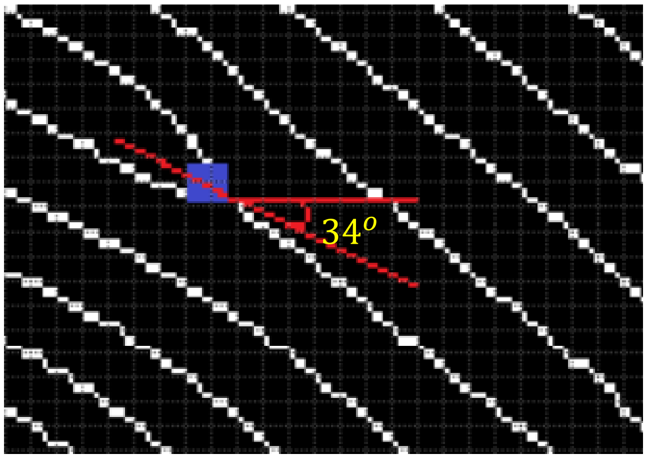
CN = 3 ⟶ Ridge Bifurcation

CN = 1 ⟶ Ridge Ending

# Constructing the Digital Template

The ridge angle of first minutiae point has been measured clock-wise from the horizontal axis





DIGITAL TEMPLATE OF INDIVIDUAL MINUTIAE POINTS REPRESENTING HARDWARE SECURITY CONSTRAINTS

| x, y | m | Minutiae type name | d | Minutiae position in binary (digital template representing hardware constraints) $Z = (x \Vert y \Vert m \Vert d)$ |
|---|---|---|---|---|
| 190, 45 | 3 | bifurcation | 34 | 10111110101101111100010 |
| 139, 46 | 3 | bifurcation | 9 | 10001011101110111001 |
| 126, 54 | 1 | ending | 189 | 1111110110110110111101 |
| 79, 64 | 1 | ending | 327 | 10011111000000110100 0111 |
| 181, 83 | 1 | ending | 38 | 1011010110100111100110 |
| 219, 84 | 3 | bifurcation | 225 | 1101101110101001111100001 |
| 159, 98 | 1 | ending | 214 | 100111111100010111010110 |
| 136, 110 | 1 | ending | 15 | 10001000110111011111 |
| 118, 115 | 1 | ending | 334 | 111011011100111101001110 |
| 248, 130 | 3 | bifurcation | 50 | 111110001000001011110010 |
| 192, 134 | 1 | ending | 46 | 1100000010000110110 1110 |
| 117, 137 | 1 | ending | 135 | 111010110001001110000111 |
| 132, 150 | 3 | bifurcation | 138 | 10000100100101101110001010 |
| 111, 164 | 1 | ending | 267 | 11011111010 01001100001011 |
| 149, 169 | 1 | ending | 239 | 1001010110101001111101111 |

X co-ordinate(x)

Y co-ordinate(y)

Minutiae type(m)

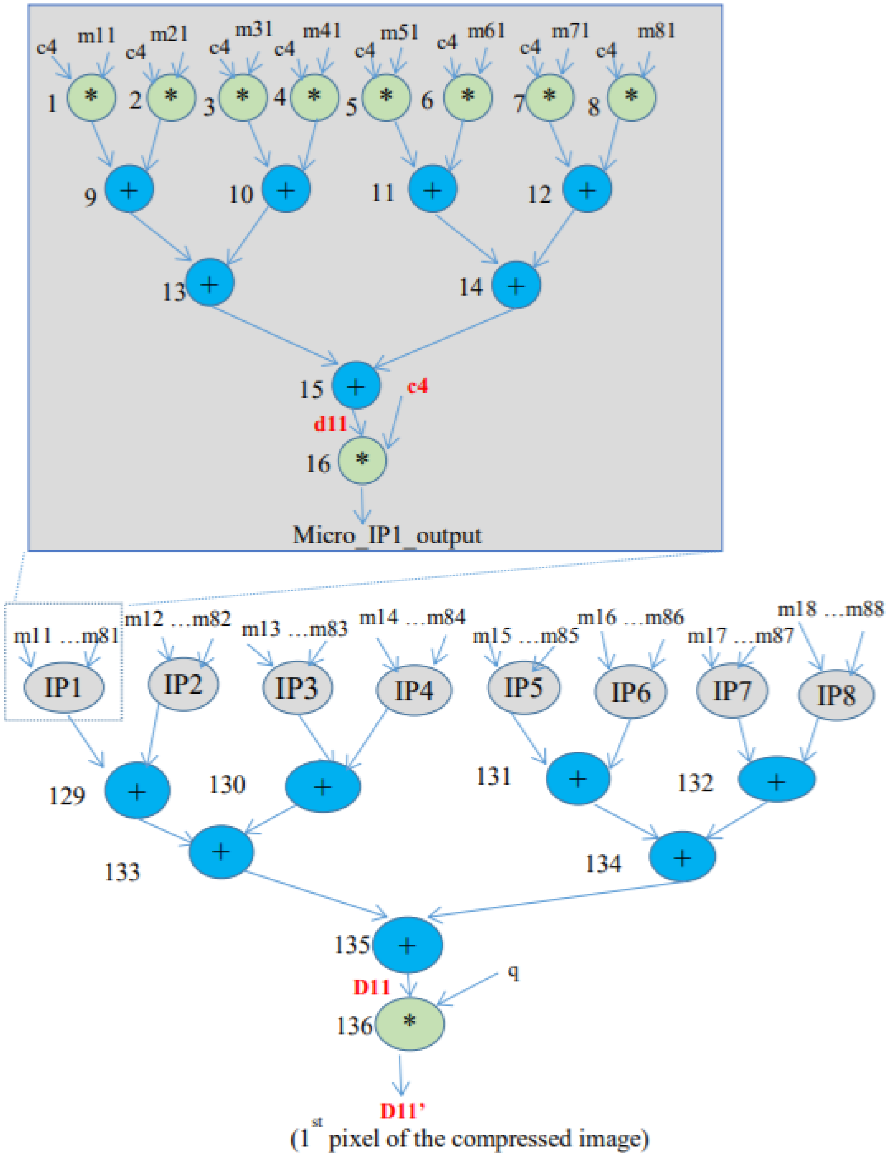
Angle(d)

Digital template obtained by concatenating templates of all 15 minutiae points

101111101011011111000101000101110111011100111111101101101101111011001111100000001101
0001111011010110100111100110110110111010100111110000110011111110001011101011010001
0001101110111111110110111001111010011101111100010000010111100101100000010000110110
1110111010110001001110000111100001001001011011100010101101111101001001100001011100
1010110101001111101111

350 bits, 148 zeros and 202 ones

# JPEG CODEC Compression Algorithm



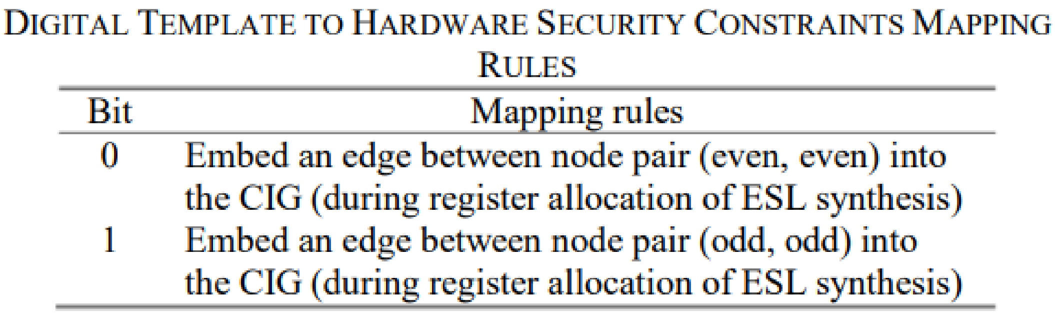
Deduced DFG of JPEG CODEC hardware accelerator

$$\begin{bmatrix}
C_4 & C_4 & C_4 & C_4 & C_4 & C_4 & C_4 & C_4 \\
C_1 & C_3 & C_5 & C_7 & -C_7 & -C_5 & -C_3 & -C_1 \\
C_2 & C_6 & -C_6 & -C_2 & -C_2 & -C_6 & -C_6 & C_2 \\
C_3 & -C_7 & -C_1 & -C_5 & C_5 & C_1 & C_7 & -C_3 \\
C_4 & -C_4 & -C_4 & C_4 & C_4 & -C_4 & -C_4 & C_4 \\
C_5 & -C_1 & C_7 & C_3 & -C_3 & -C_7 & C_1 & -C_5 \\
C_6 & -C_2 & C_2 & -C_6 & -C_6 & C_2 & -C_2 & C_6 \\
C_7 & -C_5 & C_3 & -C_1 & C_1 & -C_3 & C_5 & -C_7
\end{bmatrix}$$

. 2D- DCT coefficient matrix ‘C’

- Post multiplying the 1st pixel (D11) of DCT transformed matrix with the quantization coefficient ‘q’, the 1st pixel (D11’) of the compressed image is generated.
- Post quantization, the quantized image is converted to 1-D array using zigzag scanning and run length encoding is applied to obtain bit stream of the compressed image for storage purpose.
- In order to reconstruct the original image from the stored compressed image data, run length decoding followed by inverse zigzag scanning, inverse quantization and inverse DCT transformation are performed in the JPEG decompression process.
- The JPEG compression hardware comprises of 8 micro-IPs underneath. Each micro-IP performs 16 operations and total operations in the entire JPEG CODEC IP core are 136.

# Embedding Security Constraints into JPEG CODEC

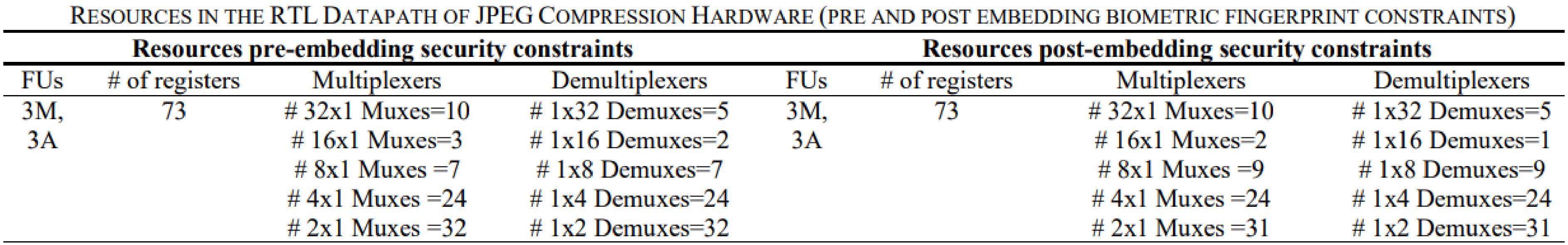


DIGITAL TEMPLATE TO HARDWARE SECURITY CONSTRAINTS MAPPING RULES

| Bit | Mapping rules |
|---|---|
| 0 | Embed an edge between node pair (even, even) into the CIG (during register allocation of ESL synthesis) |
| 1 | Embed an edge between node pair (odd, odd) into the CIG (during register allocation of ESL synthesis) |

148 zeroes → 148 even-even edges → (0,2), (0,4), . . . (0,208), (2,4), (2,6), . . . . . . (2,88), (2,90)

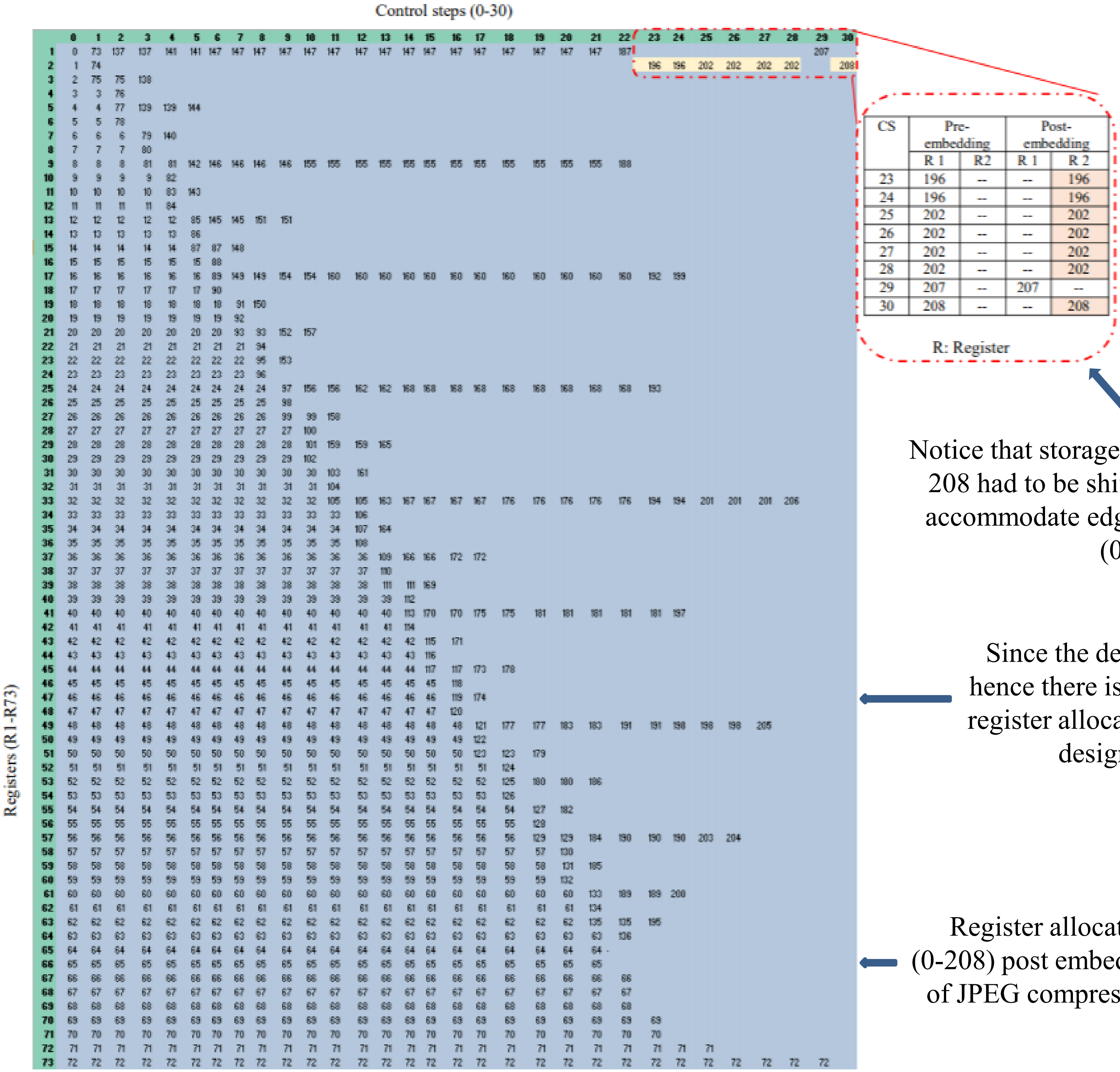
202 ones → 202 odd-odd edges → (1,3), (1,5), . . . (1,207), (3,5), (3,7), . . . . . . (3,199), (3,201)



RESOURCES IN THE RTL DATAPATH OF JPEG COMPRESSION HARDWARE (PRE AND POST EMBEDDING BIOMETRIC FINGERPRINT CONSTRAINTS)

| Resources pre-embedding security constraints | | | | Resources post-embedding security constraints | | | |
|---|---|---|---|---|---|---|---|
| FUs | # of registers | Multiplexers | Demultiplexers | FUs | # of registers | Multiplexers | Demultiplexers |
| 3M, 3A | 73 | # 32x1 Muxes=10 | # 1x32 Demuxes=5 | 3M, 3A | 73 | # 32x1 Muxes=10 | # 1x32 Demuxes=5 |
| | | # 16x1 Muxes=3 | # 1x16 Demuxes=2 | | | # 16x1 Muxes=2 | # 1x16 Demuxes=1 |
| | | # 8x1 Muxes =7 | # 1x8 Demuxes=7 | | | # 8x1 Muxes =9 | # 1x8 Demuxes=9 |
| | | # 4x1 Muxes =24 | # 1x4 Demuxes=24 | | | # 4x1 Muxes =24 | # 1x4 Demuxes=24 |
| | | # 2x1 Muxes =32 | # 1x2 Demuxes=32 | | | # 2x1 Muxes =31 | # 1x2 Demuxes=31 |

As JPEG CODEC is a huge hardware accelerator, there is no overhead of registers.



Control steps (0-30)

Registers (R1-R73)

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 73 | 137 | 137 | 141 | 141 | 147 | 147 | 147 | 147 | 147 | 147 | 147 | 147 | 147 | 147 | 147 | 147 | 147 | 147 | 147 | 147 | 187 | | | | | | | 207 | |
| 2 | 1 | 74 | | | | | | | | | | | | | | | | | | | | | | 196 | 196 | 202 | 202 | 202 | 202 | | 208 |
| 3 | 2 | 75 | 75 | 138 | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 4 | 3 | 3 | 76 | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 5 | 4 | 4 | 77 | 139 | 139 | 144 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 6 | 5 | 5 | 78 | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 7 | 6 | 6 | 6 | 79 | 140 | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 8 | 7 | 7 | 7 | 80 | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 9 | 8 | 8 | 8 | 81 | 81 | 142 | 146 | 146 | 146 | 146 | 155 | 155 | 155 | 155 | 155 | 155 | 155 | 155 | 155 | 155 | 155 | 155 | 188 | | | | | | | | |
| 10 | 9 | 9 | 9 | 9 | 82 | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 11 | 10 | 10 | 10 | 10 | 83 | 143 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 12 | 11 | 11 | 11 | 11 | 84 | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 13 | 12 | 12 | 12 | 12 | 12 | 85 | 145 | 145 | 151 | 151 | | | | | | | | | | | | | | | | | | | | | |
| 14 | 13 | 13 | 13 | 13 | 13 | 86 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 15 | 14 | 14 | 14 | 14 | 14 | 87 | 87 | 148 | | | | | | | | | | | | | | | | | | | | | | | |
| 16 | 15 | 15 | 15 | 15 | 15 | 15 | 88 | | | | | | | | | | | | | | | | | | | | | | | | |
| 17 | 16 | 16 | 16 | 16 | 16 | 16 | 89 | 149 | 149 | 154 | 154 | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 160 | 192 | 199 | | | | | | |
| 18 | 17 | 17 | 17 | 17 | 17 | 17 | 90 | | | | | | | | | | | | | | | | | | | | | | | | |
| 19 | 18 | 18 | 18 | 18 | 18 | 18 | 18 | 91 | 150 | | | | | | | | | | | | | | | | | | | | | | |
| 20 | 19 | 19 | 19 | 19 | 19 | 19 | 19 | 92 | | | | | | | | | | | | | | | | | | | | | | | |
| 21 | 20 | 20 | 20 | 20 | 20 | 20 | 20 | 93 | 93 | 152 | 157 | | | | | | | | | | | | | | | | | | | | |
| 22 | 21 | 21 | 21 | 21 | 21 | 21 | 21 | 21 | 94 | | | | | | | | | | | | | | | | | | | | | | |
| 23 | 22 | 22 | 22 | 22 | 22 | 22 | 22 | 22 | 95 | 153 | | | | | | | | | | | | | | | | | | | | | |
| 24 | 23 | 23 | 23 | 23 | 23 | 23 | 23 | 23 | 96 | | | | | | | | | | | | | | | | | | | | | | |
| 25 | 24 | 24 | 24 | 24 | 24 | 24 | 24 | 24 | 24 | 97 | 156 | 156 | 162 | 162 | 168 | 168 | 168 | 168 | 168 | 168 | 168 | 168 | 168 | 193 | | | | | | | |
| 26 | 25 | 25 | 25 | 25 | 25 | 25 | 25 | 25 | 25 | 98 | | | | | | | | | | | | | | | | | | | | | |
| 27 | 26 | 26 | 26 | 26 | 26 | 26 | 26 | 26 | 26 | 99 | 99 | 158 | | | | | | | | | | | | | | | | | | | |
| 28 | 27 | 27 | 27 | 27 | 27 | 27 | 27 | 27 | 27 | 27 | 100 | | | | | | | | | | | | | | | | | | | | |
| 29 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 101 | 159 | 159 | 165 | | | | | | | | | | | | | | | | | |
| 30 | 29 | 29 | 29 | 29 | 29 | 29 | 29 | 29 | 29 | 29 | 102 | | | | | | | | | | | | | | | | | | | | |
| 31 | 30 | 30 | 30 | 30 | 30 | 30 | 30 | 30 | 30 | 30 | 30 | 103 | 161 | | | | | | | | | | | | | | | | | | |
| 32 | 31 | 31 | 31 | 31 | 31 | 31 | 31 | 31 | 31 | 31 | 31 | 104 | | | | | | | | | | | | | | | | | | | |
| 33 | 32 | 32 | 32 | 32 | 32 | 32 | 32 | 32 | 32 | 32 | 32 | 105 | 105 | 163 | 167 | 167 | 167 | 167 | 176 | 176 | 176 | 176 | 176 | 194 | 194 | 201 | 201 | 201 | 206 | | |
| 34 | 33 | 33 | 33 | 33 | 33 | 33 | 33 | 33 | 33 | 33 | 33 | 33 | 106 | | | | | | | | | | | | | | | | | | |
| 35 | 34 | 34 | 34 | 34 | 34 | 34 | 34 | 34 | 34 | 34 | 34 | 34 | 107 | 164 | | | | | | | | | | | | | | | | | |
| 36 | 35 | 35 | 35 | 35 | 35 | 35 | 35 | 35 | 35 | 35 | 35 | 35 | 108 | | | | | | | | | | | | | | | | | | |
| 37 | 36 | 36 | 36 | 36 | 36 | 36 | 36 | 36 | 36 | 36 | 36 | 36 | 36 | 109 | 166 | 166 | 172 | 172 | | | | | | | | | | | | | |
| 38 | 37 | 37 | 37 | 37 | 37 | 37 | 37 | 37 | 37 | 37 | 37 | 37 | 37 | 110 | | | | | | | | | | | | | | | | | |
| 39 | 38 | 38 | 38 | 38 | 38 | 38 | 38 | 38 | 38 | 38 | 38 | 38 | 38 | 111 | 111 | 169 | | | | | | | | | | | | | | | |
| 40 | 39 | 39 | 39 | 39 | 39 | 39 | 39 | 39 | 39 | 39 | 39 | 39 | 39 | 39 | 112 | | | | | | | | | | | | | | | | |
| 41 | 40 | 40 | 40 | 40 | 40 | 40 | 40 | 40 | 40 | 40 | 40 | 40 | 40 | 40 | 113 | 170 | 170 | 175 | 175 | 181 | 181 | 181 | 181 | 181 | 197 | | | | | | |
| 42 | 41 | 41 | 41 | 41 | 41 | 41 | 41 | 41 | 41 | 41 | 41 | 41 | 41 | 41 | 114 | | | | | | | | | | | | | | | | |
| 43 | 42 | 42 | 42 | 42 | 42 | 42 | 42 | 42 | 42 | 42 | 42 | 42 | 42 | 42 | 42 | 115 | 171 | | | | | | | | | | | | | | |
| 44 | 43 | 43 | 43 | 43 | 43 | 43 | 43 | 43 | 43 | 43 | 43 | 43 | 43 | 43 | 43 | 116 | | | | | | | | | | | | | | | |
| 45 | 44 | 44 | 44 | 44 | 44 | 44 | 44 | 44 | 44 | 44 | 44 | 44 | 44 | 44 | 44 | 117 | 117 | 173 | 178 | | | | | | | | | | | | |
| 46 | 45 | 45 | 45 | 45 | 45 | 45 | 45 | 45 | 45 | 45 | 45 | 45 | 45 | 45 | 45 | 45 | 118 | | | | | | | | | | | | | | |
| 47 | 46 | 46 | 46 | 46 | 46 | 46 | 46 | 46 | 46 | 46 | 46 | 46 | 46 | 46 | 46 | 46 | 119 | 174 | | | | | | | | | | | | | |
| 48 | 47 | 47 | 47 | 47 | 47 | 47 | 47 | 47 | 47 | 47 | 47 | 47 | 47 | 47 | 47 | 47 | 120 | | | | | | | | | | | | | | |
| 49 | 48 | 48 | 48 | 48 | 48 | 48 | 48 | 48 | 48 | 48 | 48 | 48 | 48 | 48 | 48 | 48 | 48 | 121 | 177 | 177 | 183 | 183 | 191 | 191 | 198 | 198 | 198 | 205 | | | |
| 50 | 49 | 49 | 49 | 49 | 49 | 49 | 49 | 49 | 49 | 49 | 49 | 49 | 49 | 49 | 49 | 49 | 49 | 122 | | | | | | | | | | | | | |
| 51 | 50 | 50 | 50 | 50 | 50 | 50 | 50 | 50 | 50 | 50 | 50 | 50 | 50 | 50 | 50 | 50 | 50 | 50 | 123 | 179 | | | | | | | | | | | |
| 52 | 51 | 51 | 51 | 51 | 51 | 51 | 51 | 51 | 51 | 51 | 51 | 51 | 51 | 51 | 51 | 51 | 51 | 51 | 124 | | | | | | | | | | | | |
| 53 | 52 | 52 | 52 | 52 | 52 | 52 | 52 | 52 | 52 | 52 | 52 | 52 | 52 | 52 | 52 | 52 | 52 | 52 | 125 | 180 | 180 | 186 | | | | | | | | | |
| 54 | 53 | 53 | 53 | 53 | 53 | 53 | 53 | 53 | 53 | 53 | 53 | 53 | 53 | 53 | 53 | 53 | 53 | 53 | 126 | | | | | | | | | | | | |
| 55 | 54 | 54 | 54 | 54 | 54 | 54 | 54 | 54 | 54 | 54 | 54 | 54 | 54 | 54 | 54 | 54 | 54 | 54 | 54 | 127 | 182 | | | | | | | | | | |
| 56 | 55 | 55 | 55 | 55 | 55 | 55 | 55 | 55 | 55 | 55 | 55 | 55 | 55 | 55 | 55 | 55 | 55 | 55 | 55 | 128 | | | | | | | | | | | |
| 57 | 56 | 56 | 56 | 56 | 56 | 56 | 56 | 56 | 56 | 56 | 56 | 56 | 56 | 56 | 56 | 56 | 56 | 56 | 56 | 129 | 129 | 184 | 190 | 190 | 190 | 203 | 204 | | | | |
| 58 | 57 | 57 | 57 | 57 | 57 | 57 | 57 | 57 | 57 | 57 | 57 | 57 | 57 | 57 | 57 | 57 | 57 | 57 | 57 | 57 | 130 | | | | | | | | | | |
| 59 | 58 | 58 | 58 | 58 | 58 | 58 | 58 | 58 | 58 | 58 | 58 | 58 | 58 | 58 | 58 | 58 | 58 | 58 | 58 | 58 | 131 | 185 | | | | | | | | | |
| 60 | 59 | 59 | 59 | 59 | 59 | 59 | 59 | 59 | 59 | 59 | 59 | 59 | 59 | 59 | 59 | 59 | 59 | 59 | 59 | 59 | 132 | | | | | | | | | | |
| 61 | 60 | 60 | 60 | 60 | 60 | 60 | 60 | 60 | 60 | 60 | 60 | 60 | 60 | 60 | 60 | 60 | 60 | 60 | 60 | 60 | 60 | 133 | 189 | 189 | 200 | | | | | | |
| 62 | 61 | 61 | 61 | 61 | 61 | 61 | 61 | 61 | 61 | 61 | 61 | 61 | 61 | 61 | 61 | 61 | 61 | 61 | 61 | 61 | 61 | 134 | | | | | | | | | |
| 63 | 62 | 62 | 62 | 62 | 62 | 62 | 62 | 62 | 62 | 62 | 62 | 62 | 62 | 62 | 62 | 62 | 62 | 62 | 62 | 62 | 62 | 135 | 135 | 195 | | | | | | | |
| 64 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 63 | 136 | | | | | | | | |
| 65 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | 64 | | | | | | | | | |
| 66 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | | | | | | | | | |
| 67 | 66 | 66 | 66 | 66 | 66 | 66 | 66 | 66 | 66 | 66 | 66 | 66 | 66 | 66 | 66 | 66 | 66 | 66 | 66 | 66 | 66 | 66 | 66 | | | | | | | | |
| 68 | 67 | 67 | 67 | 67 | 67 | 67 | 67 | 67 | 67 | 67 | 67 | 67 | 67 | 67 | 67 | 67 | 67 | 67 | 67 | 67 | 67 | 67 | 67 | | | | | | | | |
| 69 | 68 | 68 | 68 | 68 | 68 | 68 | 68 | 68 | 68 | 68 | 68 | 68 | 68 | 68 | 68 | 68 | 68 | 68 | 68 | 68 | 68 | 68 | 68 | | | | | | | | |
| 70 | 69 | 69 | 69 | 69 | 69 | 69 | 69 | 69 | 69 | 69 | 69 | 69 | 69 | 69 | 69 | 69 | 69 | 69 | 69 | 69 | 69 | 69 | 69 | 69 | | | | | | | |
| 71 | 70 | 70 | 70 | 70 | 70 | 70 | 70 | 70 | 70 | 70 | 70 | 70 | 70 | 70 | 70 | 70 | 70 | 70 | 70 | 70 | 70 | 70 | 70 | 70 | | | | | | | |
| 72 | 71 | 71 | 71 | 71 | 71 | 71 | 71 | 71 | 71 | 71 | 71 | 71 | 71 | 71 | 71 | 71 | 71 | 71 | 71 | 71 | 71 | 71 | 71 | 71 | 71 | 71 | | | | | |
| 73 | 72 | 72 | 72 | 72 | 72 | 72 | 72 | 72 | 72 | 72 | 72 | 72 | 72 | 72 | 72 | 72 | 72 | 72 | 72 | 72 | 72 | 72 | 72 | 72 | 72 | 72 | 72 | 72 | 72 | 72 | |

| CS | Pre-embedding | | Post-embedding | |
|---|---|---|---|---|
| | R 1 | R2 | R 1 | R 2 |
| 23 | 196 | -- | -- | 196 |
| 24 | 196 | -- | -- | 196 |
| 25 | 202 | -- | -- | 202 |
| 26 | 202 | -- | -- | 202 |
| 27 | 202 | -- | -- | 202 |
| 28 | 202 | -- | -- | 202 |
| 29 | 207 | -- | 207 | -- |
| 30 | 208 | -- | -- | 208 |

R: Register

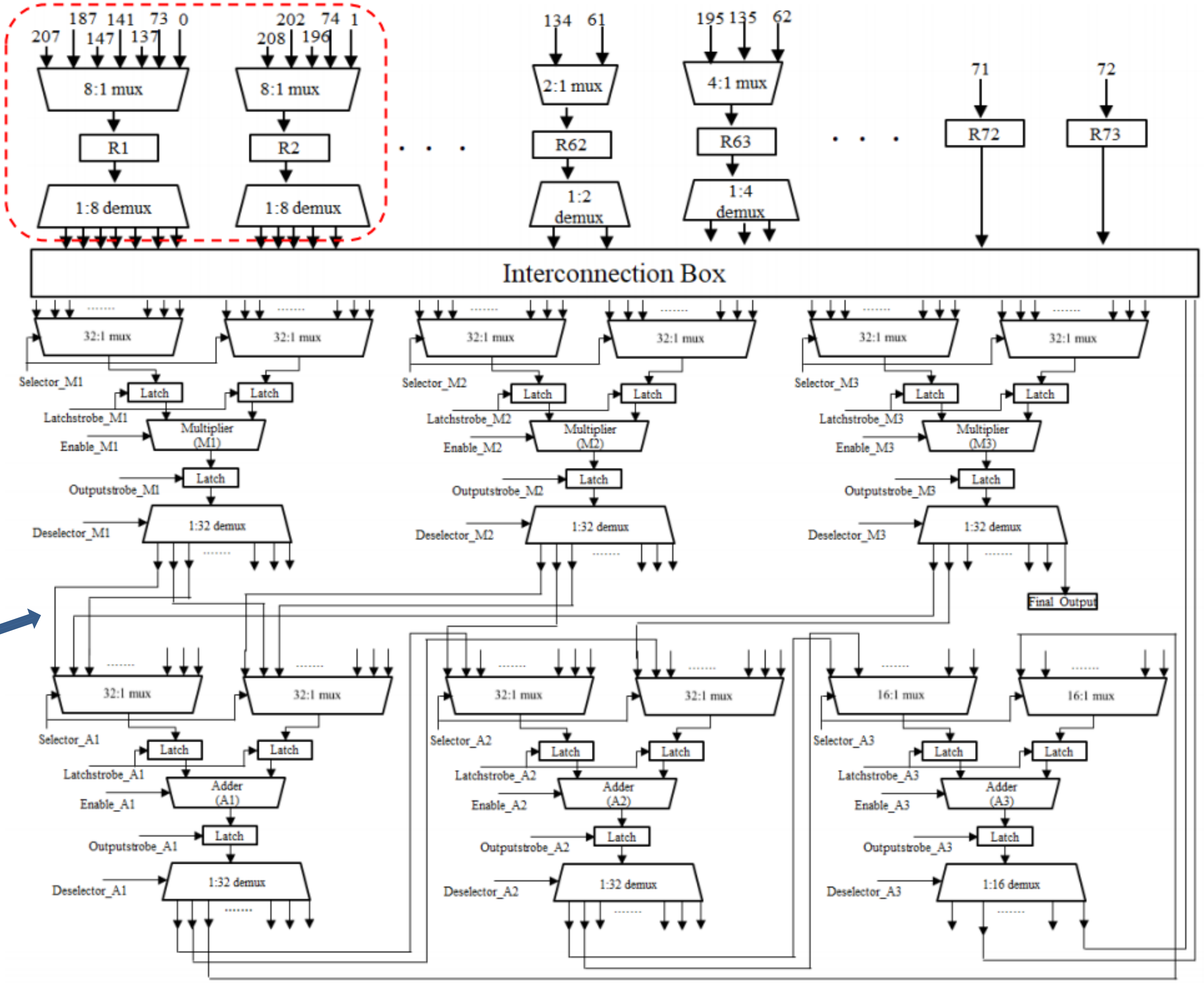
Notice that storage variables 196, 202 and 208 had to be shifted from R1 to R2 to accommodate edges (0,196), (0,202) & (0,208).

Since the design is very large, hence there isn't much change in register allocation and there is no design overhead.

Register allocation of storage variables (0-208) post embedding during ESL synthesis of JPEG compression hardware accelerator

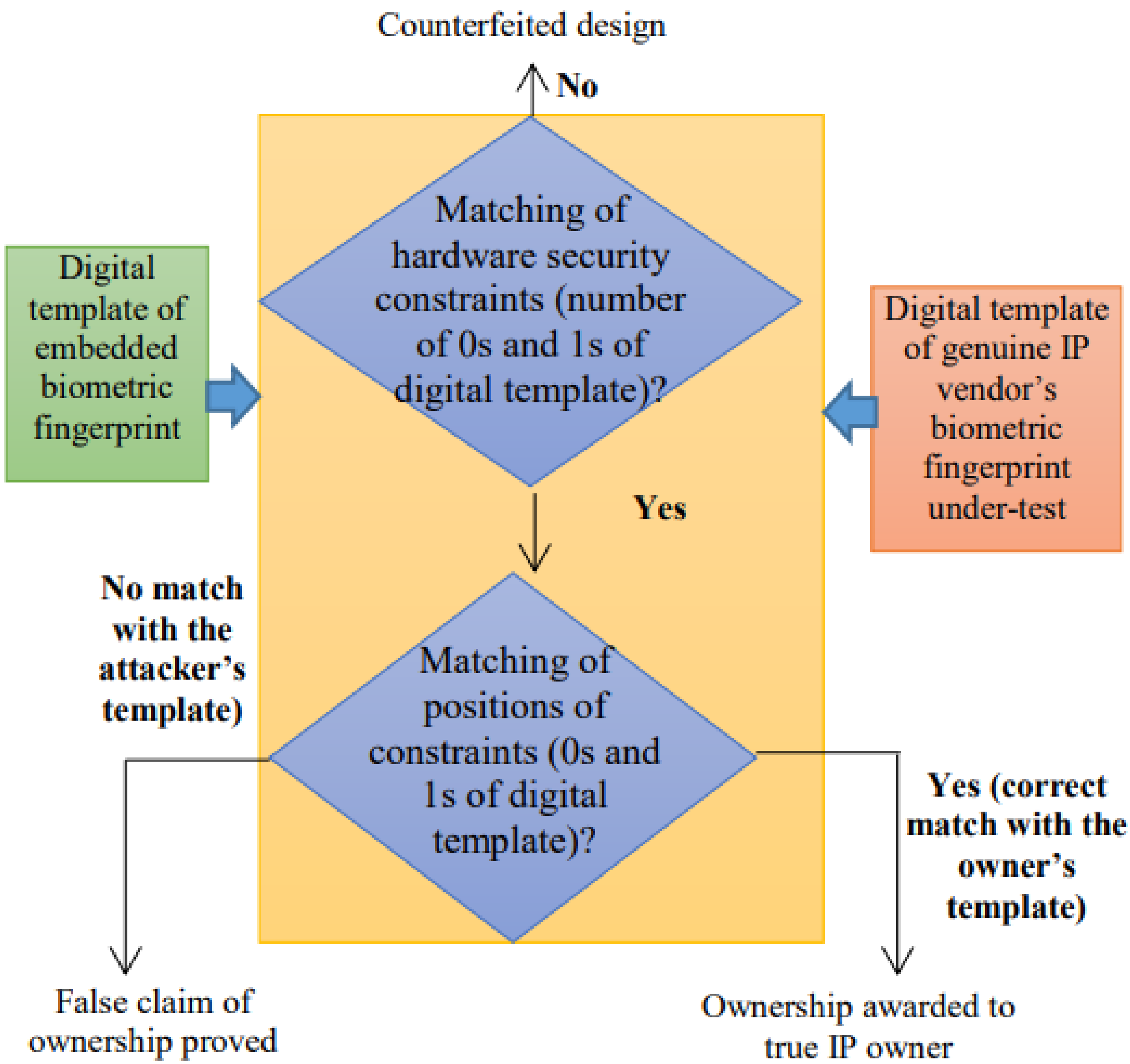
# RTL Datapath of the Secured Design

The red box highlights the changes in register allocation amongst R1 and R2.



Notice how rest of the design hardware is unaffected. Further security constraints can be added by using much larger digital signature templates.

# Detection of Biometric Fingerprint


Counterfeited design

No

Digital template of embedded biometric fingerprint

Matching of hardware security constraints (number of 0s and 1s of digital template)?

Digital template of genuine IP vendor's biometric fingerprint under-test

Yes

No match with the attacker's template)

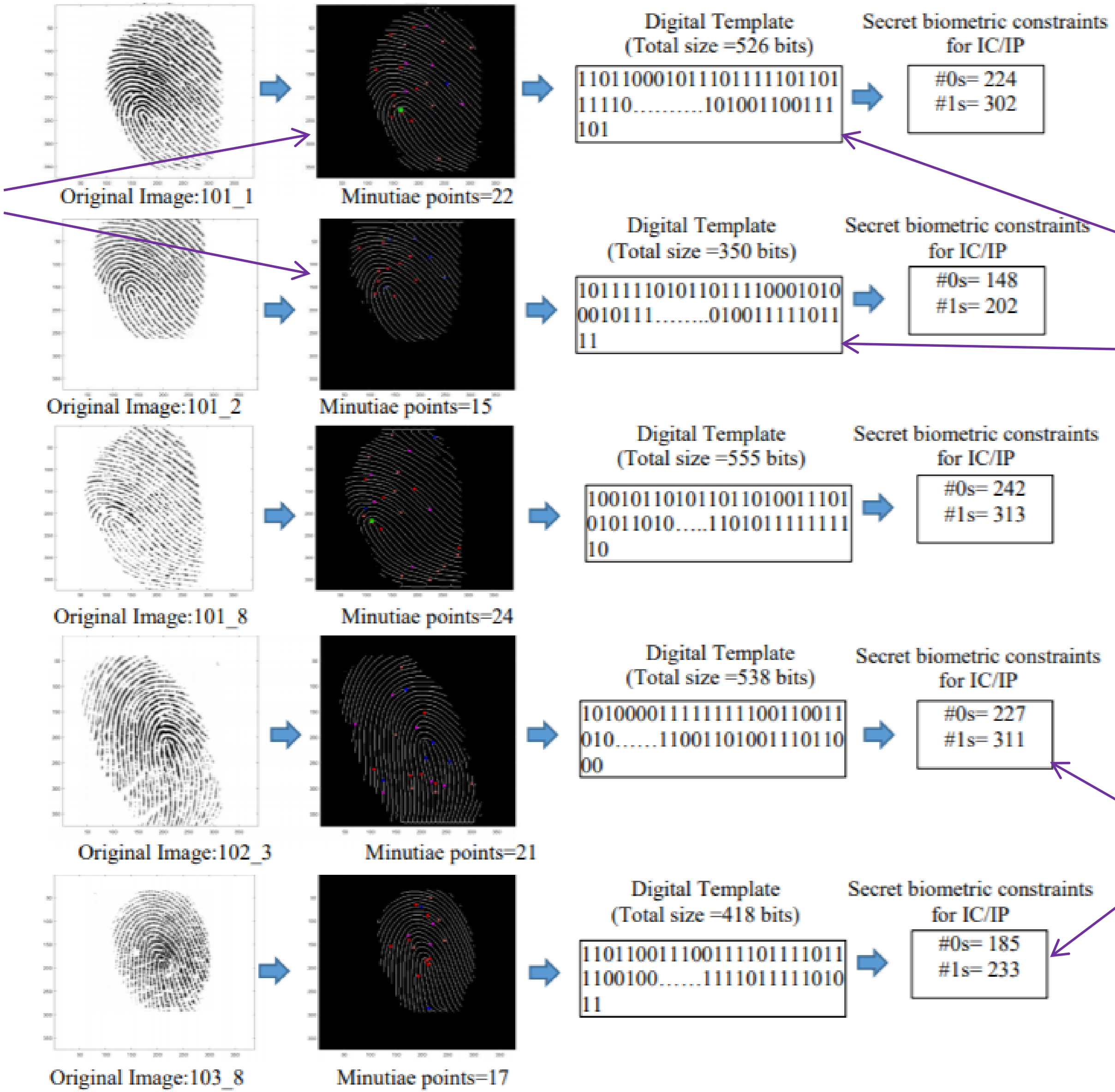
Matching of positions of constraints (0s and 1s of digital template)?

Yes (correct match with the owner's template)

False claim of ownership proved

Ownership awarded to true IP owner

# Biometric Fingerprints of 5 different people



Original Image:101_1 → Minutiae points=22 → Digital Template (Total size =526 bits)

110110001011101111101101
11110..........101001100111
101

→ Secret biometric constraints for IC/IP

#0s= 224
#1s= 302

Original Image:101_2 → Minutiae points=15 → Digital Template (Total size =350 bits)

10111111010110111110001010
0010111........0100111111011
11

→ Secret biometric constraints for IC/IP

#0s= 148
#1s= 202

Original Image:101_8 → Minutiae points=24 → Digital Template (Total size =555 bits)

100101101011011010011101
01011010.....1101011111111
10

→ Secret biometric constraints for IC/IP

#0s= 242
#1s= 313

Original Image:102_3 → Minutiae points=21 → Digital Template (Total size =538 bits)

1010000111111111100110011
010......11001101001110110
00

→ Secret biometric constraints for IC/IP

#0s= 227
#1s= 311

Original Image:103_8 → Minutiae points=17 → Digital Template (Total size =418 bits)

1101100111001111101111011
1100100......1111011111010
11

→ Secret biometric constraints for IC/IP

#0s= 185
#1s= 233

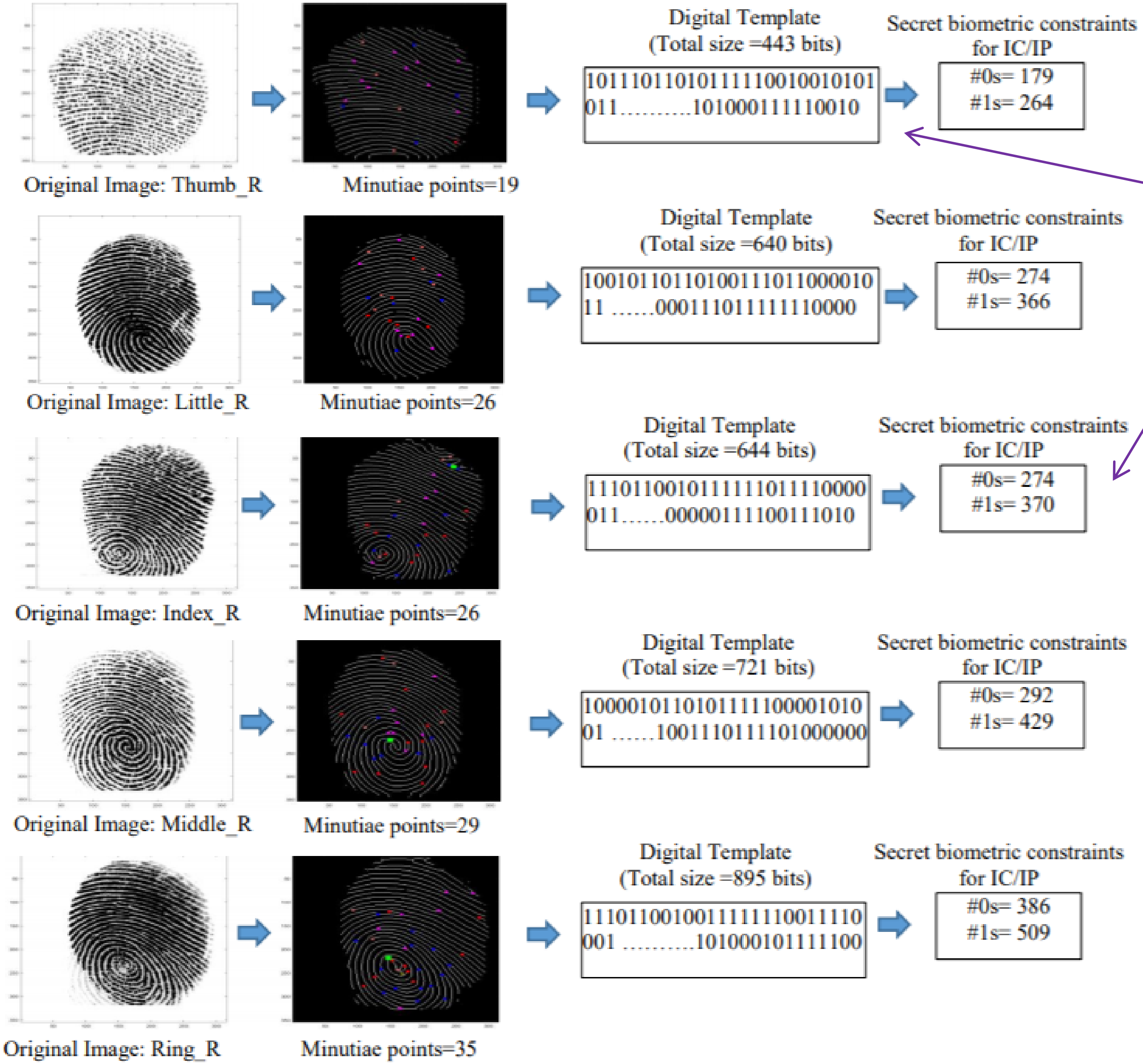
Notice how different fingerprints have different sets of minutiae points.

As fingerprints are naturally unique, there is bound to be atleast one difference at some bit position in the digital template, thereby nullifying claims for IP ownership.

Notice the varying number of bits in the digital signatures.

Notice the varying number of 0s and 1s.

# Biometric Fingerprints of 5 fingers of the same person



Original Image: Thumb_R

Minutiae points=19

Digital Template
(Total size =443 bits)

1011101101010111110010010101
011..........101000111110010

Secret biometric constraints
for IC/IP

#0s= 179
#1s= 264

Original Image: Little_R

Minutiae points=26

Digital Template
(Total size =640 bits)

1001011011010011101100000010
11 ......0001110111111110000

Secret biometric constraints
for IC/IP

#0s= 274
#1s= 366

Original Image: Index_R

Minutiae points=26

Digital Template
(Total size =644 bits)

1110110010111111011110000
011......00000111100111010

Secret biometric constraints
for IC/IP

#0s= 274
#1s= 370

Original Image: Middle_R

Minutiae points=29

Digital Template
(Total size =721 bits)

100001011010101111100001010
01 ......1001110111101000000

Secret biometric constraints
for IC/IP

#0s= 292
#1s= 429

Original Image: Ring_R

Minutiae points=35

Digital Template
(Total size =895 bits)

1110110010011111110011110
001 ...........101000101111100

Secret biometric constraints
for IC/IP

#0s= 386
#1s= 509

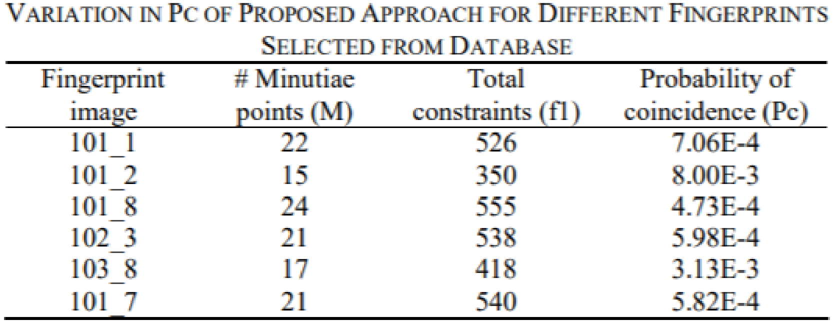
Notice how varied the fingerprints of different fingers of the same person are.

Notice that the Ring finger has the most minutiae points while the Thumb has the least. Hence, a design secured using the Ring finger is more secure.

# Security Analysis

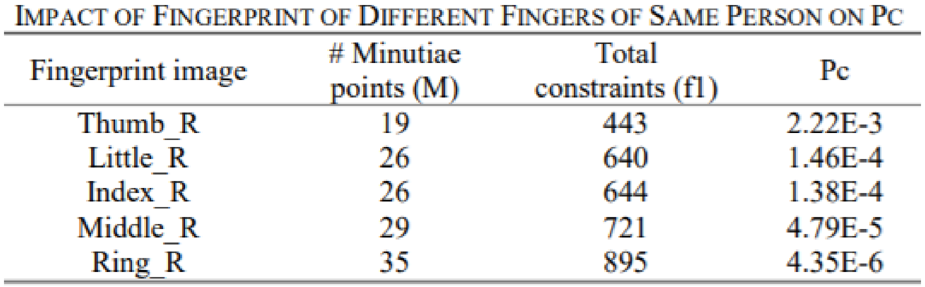
Probability of Coincidence, $Pc = \left(1-\frac{1}{R}\right)^{f1} * \left(1-\frac{1}{\pi_{i=1}^{n} N(Fi)}\right)^{f2}$

R ➡ number of colors/ registers in original CIG,
f1 ➡ number of additional edges embedded into the CIG (in register allocation phase)
f2 ➡ number of constraints embedded in the functional unit (FU) vendor allocation phase.
n total types of FU



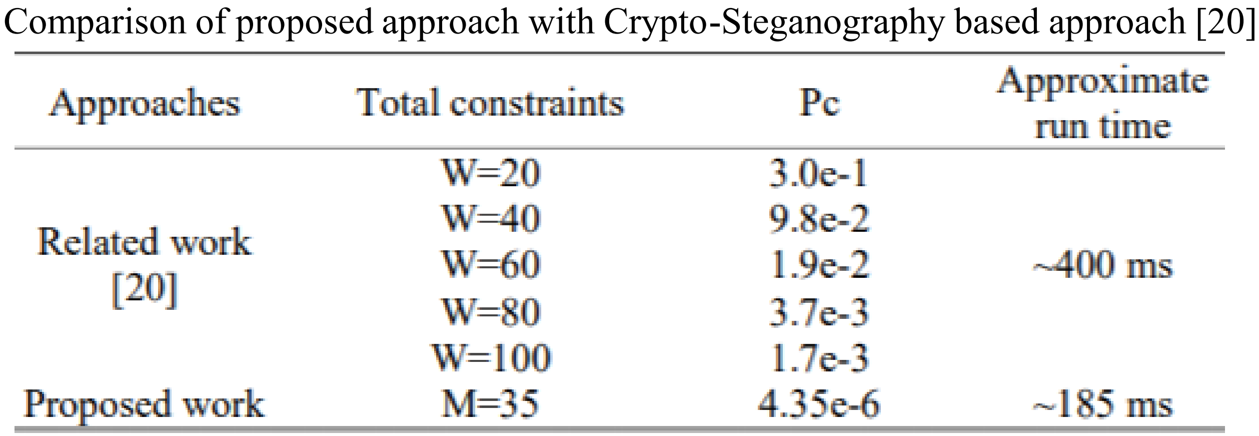
VARIATION IN PC OF PROPOSED APPROACH FOR DIFFERENT FINGERPRINTS SELECTED FROM DATABASE

| Fingerprint image | # Minutiae points (M) | Total constraints (f1) | Probability of coincidence (Pc) |
|---|---|---|---|
| 101_1 | 22 | 526 | 7.06E-4 |
| 101_2 | 15 | 350 | 8.00E-3 |
| 101_8 | 24 | 555 | 4.73E-4 |
| 102_3 | 21 | 538 | 5.98E-4 |
| 103_8 | 17 | 418 | 3.13E-3 |
| 101_7 | 21 | 540 | 5.82E-4 |



IMPACT OF FINGERPRINT OF DIFFERENT FINGERS OF SAME PERSON ON PC

| Fingerprint image | # Minutiae points (M) | Total constraints (f1) | Pc |
|---|---|---|---|
| Thumb_R | 19 | 443 | 2.22E-3 |
| Little_R | 26 | 640 | 1.46E-4 |
| Index_R | 26 | 644 | 1.38E-4 |
| Middle_R | 29 | 721 | 4.79E-5 |
| Ring_R | 35 | 895 | 4.35E-6 |



Comparison of proposed approach with Crypto-Steganography based approach [20]

| Approaches | Total constraints | Pc | Approximate run time |
|---|---|---|---|
| Related work [20] | W=20 | 3.0e-1 | ~400 ms |
| | W=40 | 9.8e-2 | |
| | W=60 | 1.9e-2 | |
| | W=80 | 3.7e-3 | |
| | W=100 | 1.7e-3 | |
| Proposed work | M=35 | 4.35e-6 | ~185 ms |

*Note: The probability of coincidence represents the probability of coincidently detecting hardware security constraints in an unsecured design. Since we don't want any coincidence between signatures of original IP vendor and adversary, it is desirable for Pc to be as low as possible.*

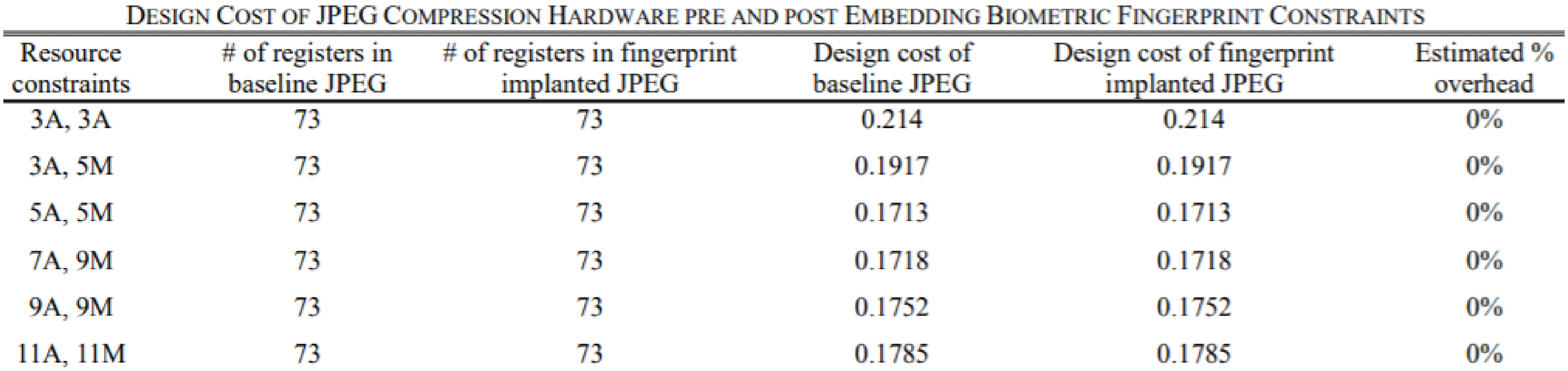
# Design Cost Analysis

Design Cost, $C_d(F_i) = w_1 \frac{D_T}{D_{max}} + w_2 \frac{A_T}{A_{max}}$

$D_T$ & $A_T$ ⟶ design delay and area respectively

$A_{max}$ and $D_{max}$ ⟶ maximum area and delay

$w_1$ and $w_2$ ⟶ user defined weights (both kept at 0.5 to assign equal preference).



Design Cost of JPEG Compression Hardware pre and post Embedding Biometric Fingerprint Constraints

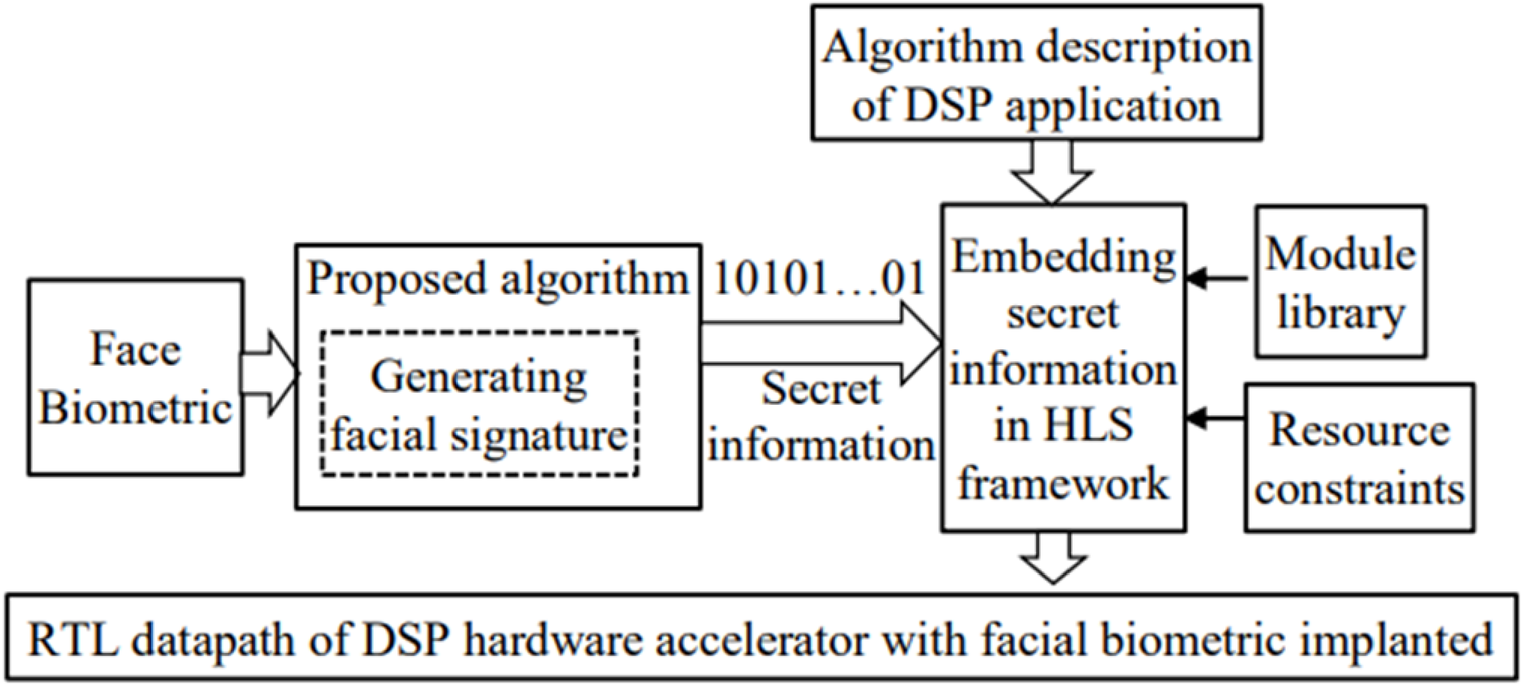
| Resource constraints | # of registers in baseline JPEG | # of registers in fingerprint implanted JPEG | Design cost of baseline JPEG | Design cost of fingerprint implanted JPEG | Estimated % overhead |
|---|---|---|---|---|---|
| 3A, 3A | 73 | 73 | 0.214 | 0.214 | 0% |
| 3A, 5M | 73 | 73 | 0.1917 | 0.1917 | 0% |
| 5A, 5M | 73 | 73 | 0.1713 | 0.1713 | 0% |
| 7A, 9M | 73 | 73 | 0.1718 | 0.1718 | 0% |
| 9A, 9M | 73 | 73 | 0.1752 | 0.1752 | 0% |
| 11A, 11M | 73 | 73 | 0.1785 | 0.1785 | 0% |

Notice how there is no design overhead for JPEG, implying that the proposed approach is apt for

# Facial Biometric for Hardware Security

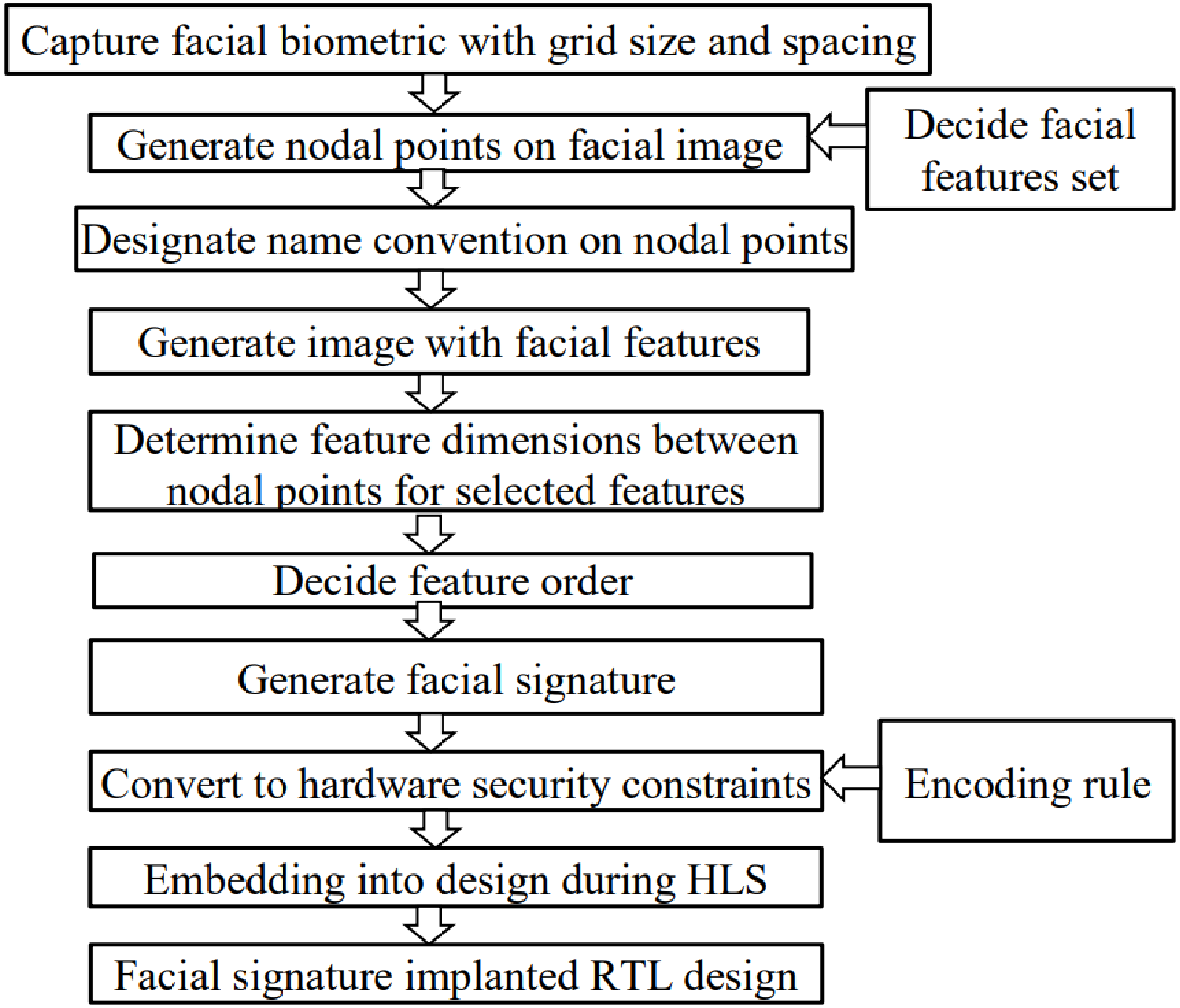
- First approach in the literature that leverages unique facial biometric information to secure hardware accelerators.
- Contact-less mechanism for securing hardware accelerators.
- Low implementation complexity and higher robustness during authentication/verification process.
- Zero dependence on external factors.
- Non-vulnerable and non-replicable technique.

# Overview of the Facial Biometric Approach



Algorithm description of DSP application
Face Biometric
Proposed algorithm
Generating facial signature
10101…01
Secret information
Embedding secret information in HLS framework
Module library
Resource constraints
RTL datapath of DSP hardware accelerator with facial biometric implanted

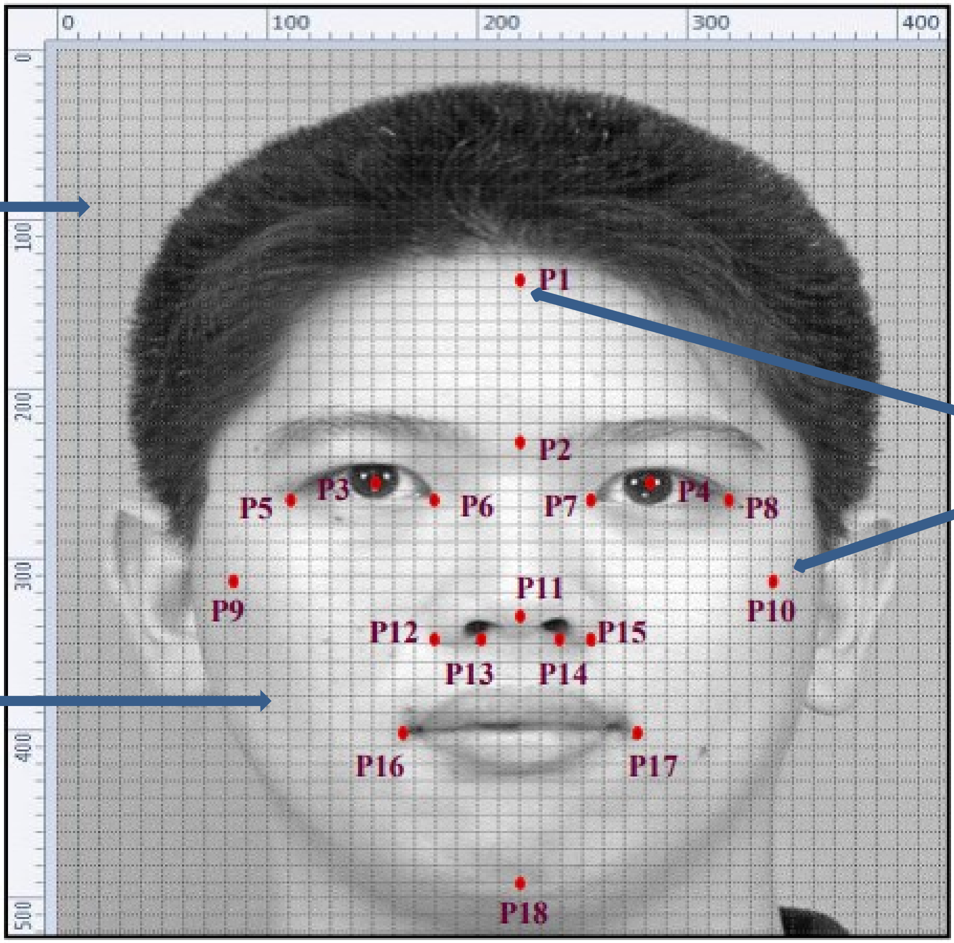
# Flow of the proposed Face Biometric Approach


Capture facial biometric with grid size and spacing
Generate nodal points on facial image
Decide facial features set
Designate name convention on nodal points
Generate image with facial features
Determine feature dimensions between nodal points for selected features
Decide feature order
Generate facial signature
Convert to hardware security constraints
Encoding rule
Embedding into design during HLS
Facial signature implanted RTL design

# Generating Nodal Points

Grid Spacing (Used to calculate the distance between nodes)

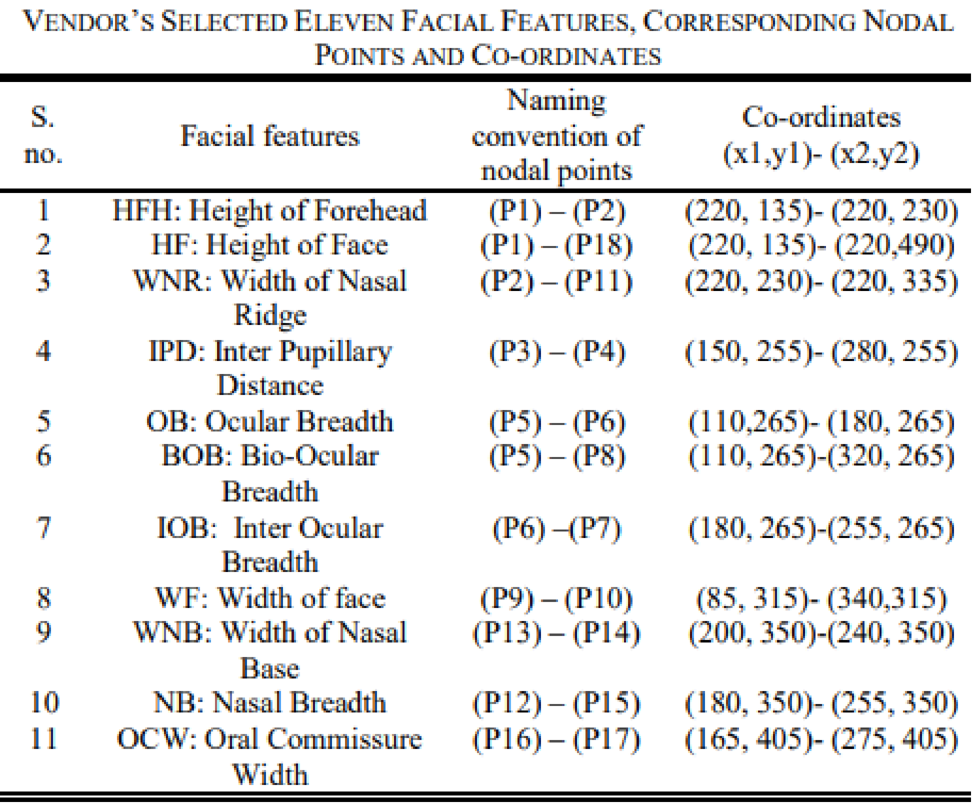
Facial Biometric



Nodal Points

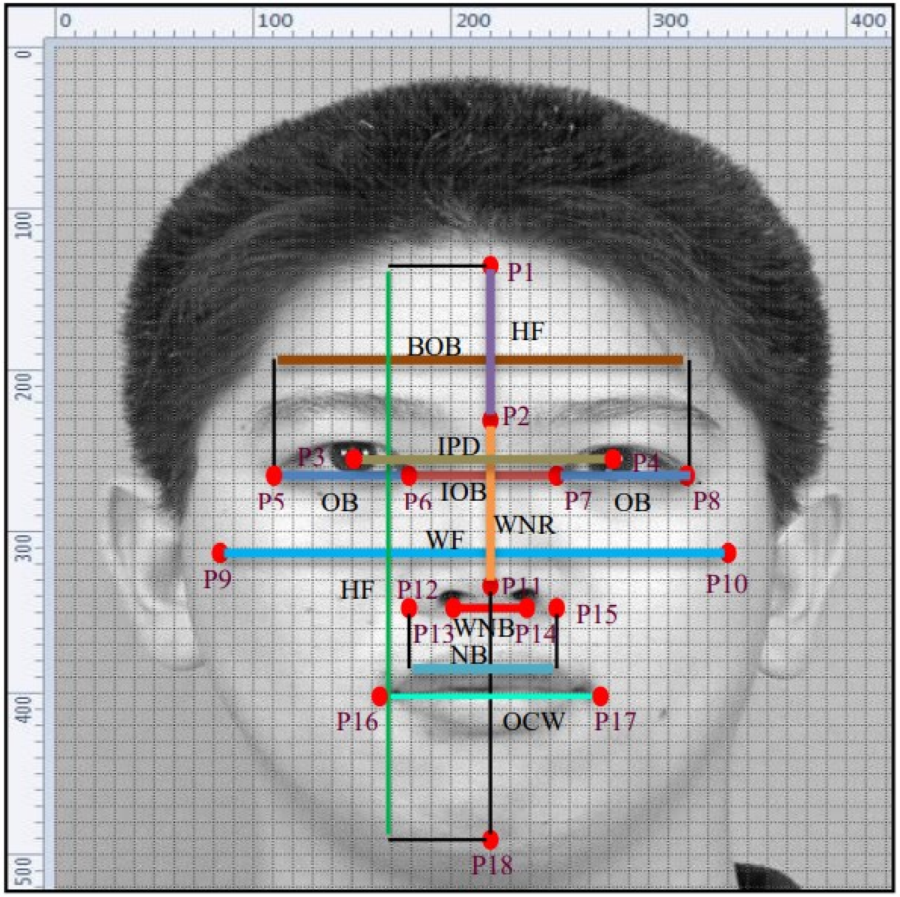
Naming the generated nodal points on the facial biometric image according to

# Determining Facial Features



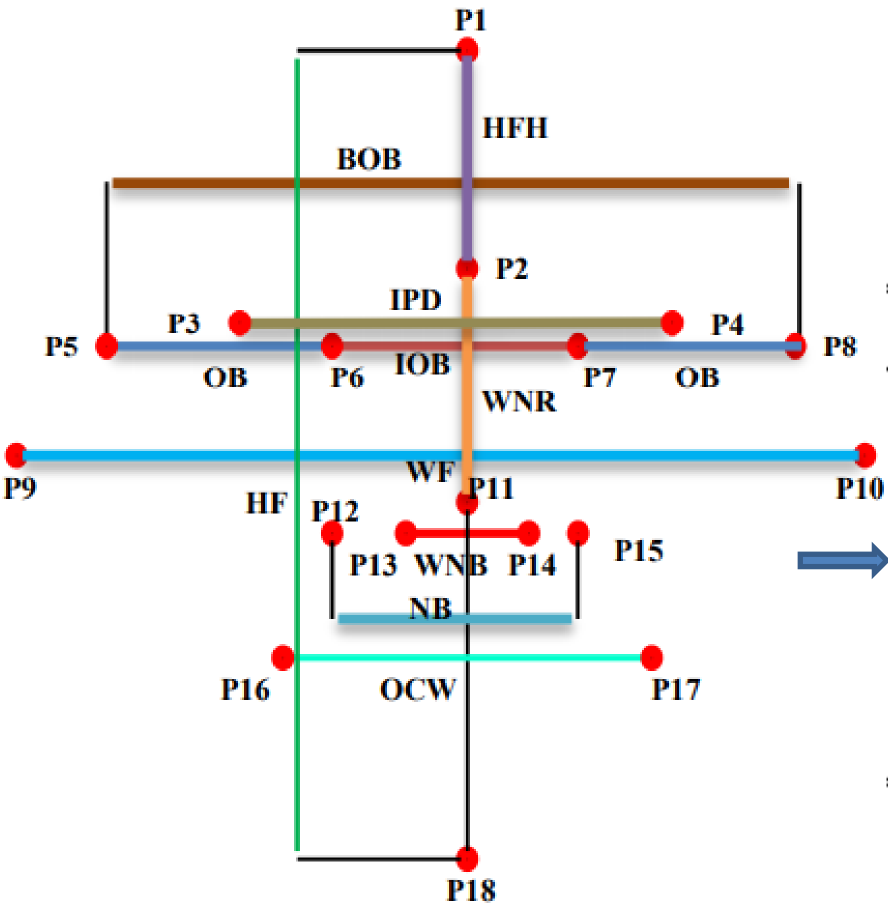
Vendor's Selected Eleven Facial Features, Corresponding Nodal Points and Co-ordinates

| S. no. | Facial features | Naming convention of nodal points | Co-ordinates (x1,y1)- (x2,y2) |
|---|---|---|---|
| 1 | HFH: Height of Forehead | (P1) – (P2) | (220, 135)- (220, 230) |
| 2 | HF: Height of Face | (P1) – (P18) | (220, 135)- (220,490) |
| 3 | WNR: Width of Nasal Ridge | (P2) – (P11) | (220, 230)- (220, 335) |
| 4 | IPD: Inter Pupillary Distance | (P3) – (P4) | (150, 255)- (280, 255) |
| 5 | OB: Ocular Breadth | (P5) – (P6) | (110,265)- (180, 265) |
| 6 | BOB: Bio-Ocular Breadth | (P5) – (P8) | (110, 265)-(320, 265) |
| 7 | IOB: Inter Ocular Breadth | (P6) –(P7) | (180, 265)-(255, 265) |
| 8 | WF: Width of face | (P9) – (P10) | (85, 315)- (340,315) |
| 9 | WNB: Width of Nasal Base | (P13) – (P14) | (200, 350)-(240, 350) |
| 10 | NB: Nasal Breadth | (P12) – (P15) | (180, 350)- (255, 350) |
| 11 | OCW: Oral Commissure Width | (P16) – (P17) | (165, 405)- (275, 405) |

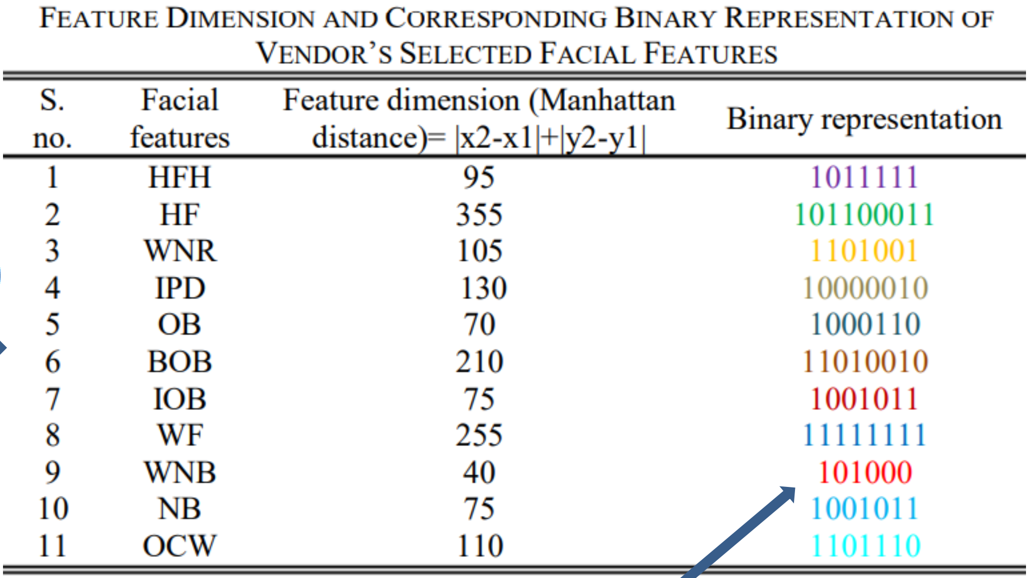


Plotting the selected features on the Facial Biometric.

# Measuring Facial Feature Dimensions and Conversion into Binary Representation



Skeleton of the vendor's selected eleven facial

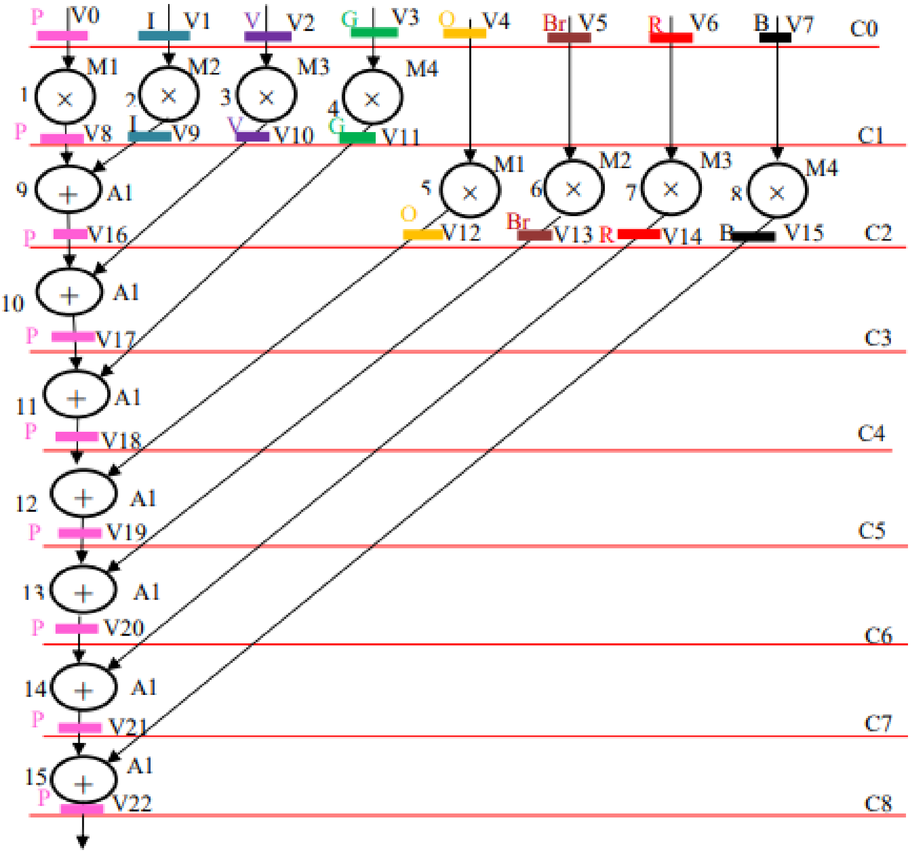


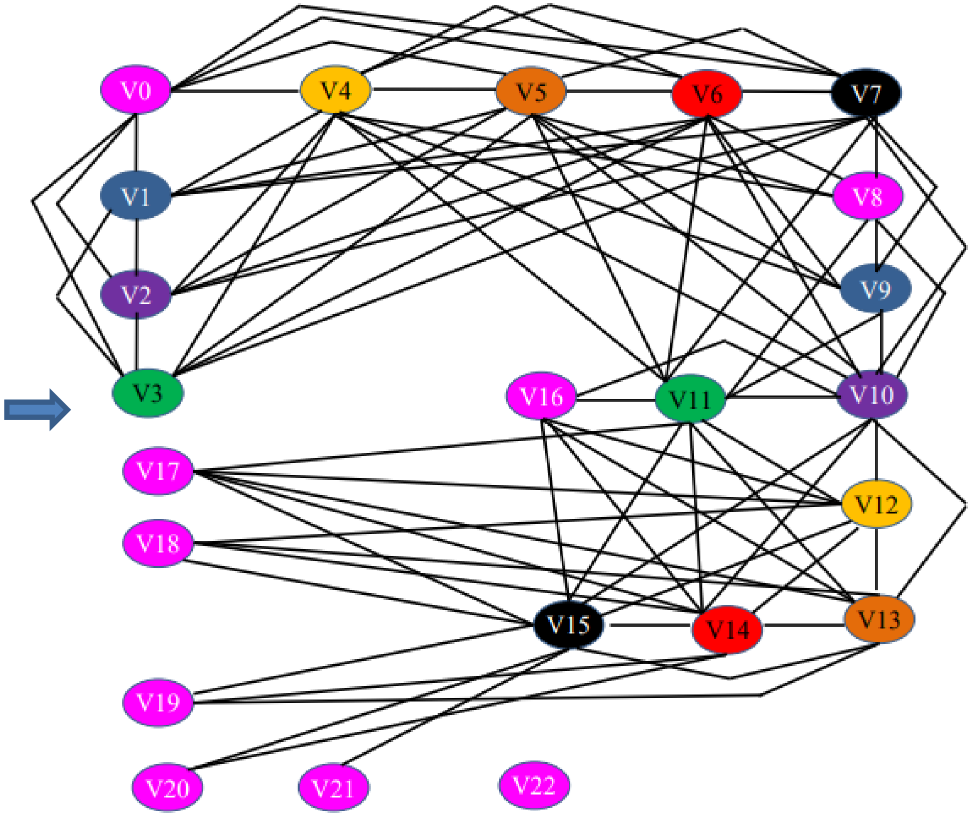
Feature Dimension and Corresponding Binary Representation of Vendor's Selected Facial Features

| S. no. | Facial features | Feature dimension (Manhattan distance)= \|x2-x1\|+\|y2-y1\| | Binary representation |
|---|---|---|---|
| 1 | HFH | 95 | 1011111 |
| 2 | HF | 355 | 101100011 |
| 3 | WNR | 105 | 1101001 |
| 4 | IPD | 130 | 10000010 |
| 5 | OB | 70 | 1000110 |
| 6 | BOB | 210 | 11010010 |
| 7 | IOB | 75 | 1001011 |
| 8 | WF | 255 | 11111111 |
| 9 | WNB | 40 | 101000 |
| 10 | NB | 75 | 1001011 |
| 11 | OCW | 110 | 1101110 |

Notice the varying lengths of feature dimensions in their binary representations. This lack of uniformity is an advantage as the adversary has no clue about the points of concatenation, making extraction of individual dimensions impossible.

# Converting Scheduled Data Flow Graph to Coloured Interval Graph





CIG showing edges between storage variables existing in the same time span.

*Notice 'V15' has most edges (10) as it shares time span with 10 storage variables and 'V22' has no edges because it doesn't share its time span with any other storage variable.*

# Deciding Feature Order and Generating Digital Template

*11 facial features*

HFH & IPD & BOB & IOB & OB & WNR & WF & HF & WNB & NB & OC

↓

101111110000010110100101001011100011011010011111111101100011101000100101111 01110

11 features, 81 bits (34 zeroes and 47 ones)

OB & HF & WNB & IPD & OC & NB & IOB & BOB & WF & HFH & WNR

↓

100011010110001110100010000010110111010010111001011110100101111111110111111101001

11 features, 81 bits (34 zeroes and 47 ones)

WNB & WNR & HFH & OC & HF & IOB

↓

1010001101001101111111011101011000111001011

6 features, 43 bits (17 zeroes and 26 ones), sufficient enough to secure small designs.

# Embedding Security Constraints into the CIG

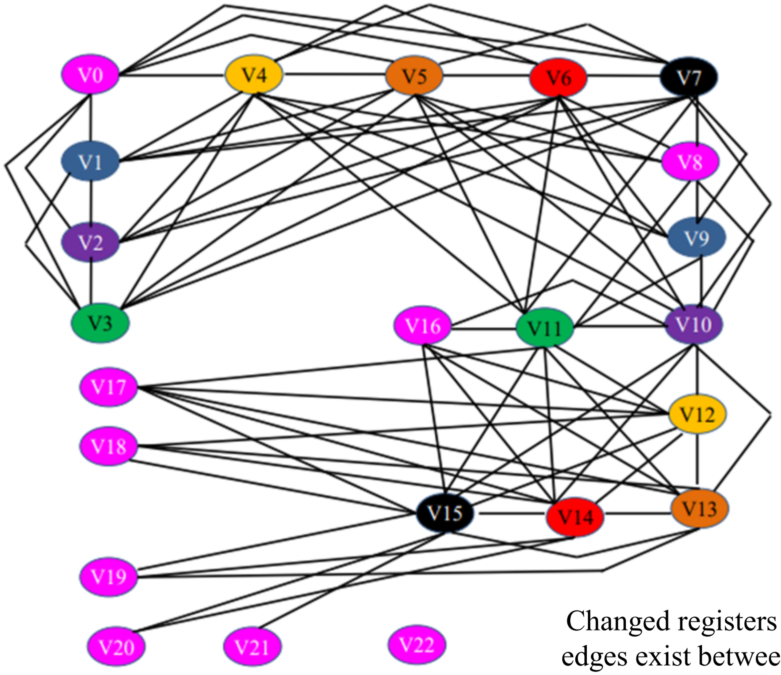
ENCODING OF FACIAL SIGNATURE INTO HARDWARE SECURITY CONSTRAINTS

| Bit | Encoding rules |
|---|---|
| 0 | Encoded as an edge between node pair (even, even) into the CIG |
| 1 | Encoded as an edge between node pair (odd, odd) into the CIG |

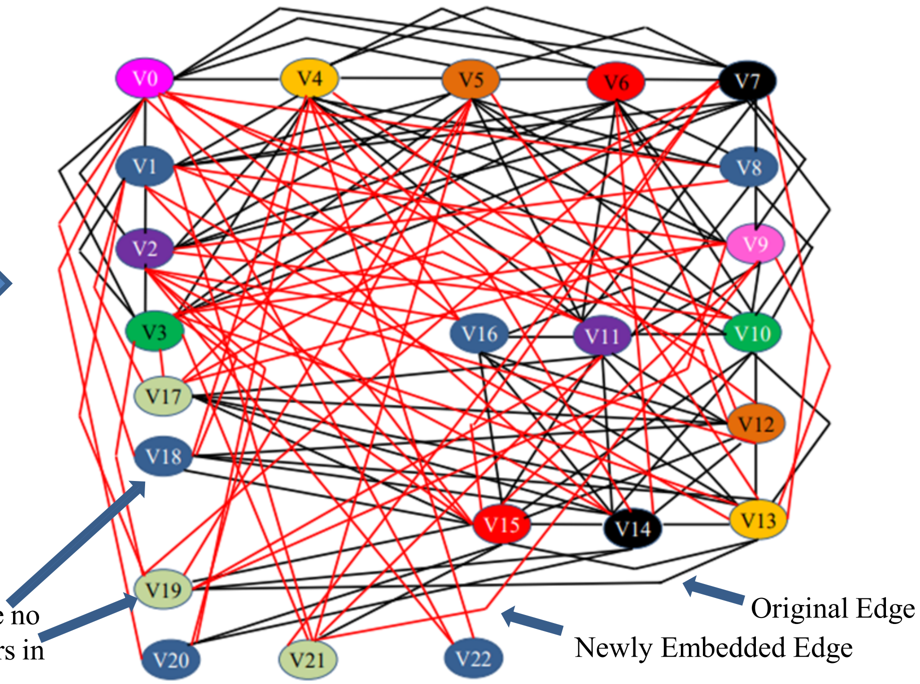
101111110000010110100101001011100011011010011111111101100011101000100101111101110

34 zeroes → 34 even-even edges → (0,2), (0,4), . . . (0,22), (2,4), (2,6), . . . . . . (6,12), (6,14)

47 ones → 47 odd-odd edges → (1,3), (1,5), . . . (1,21), (3,5), (3,7), . . . . . . (13,15), (13,17)



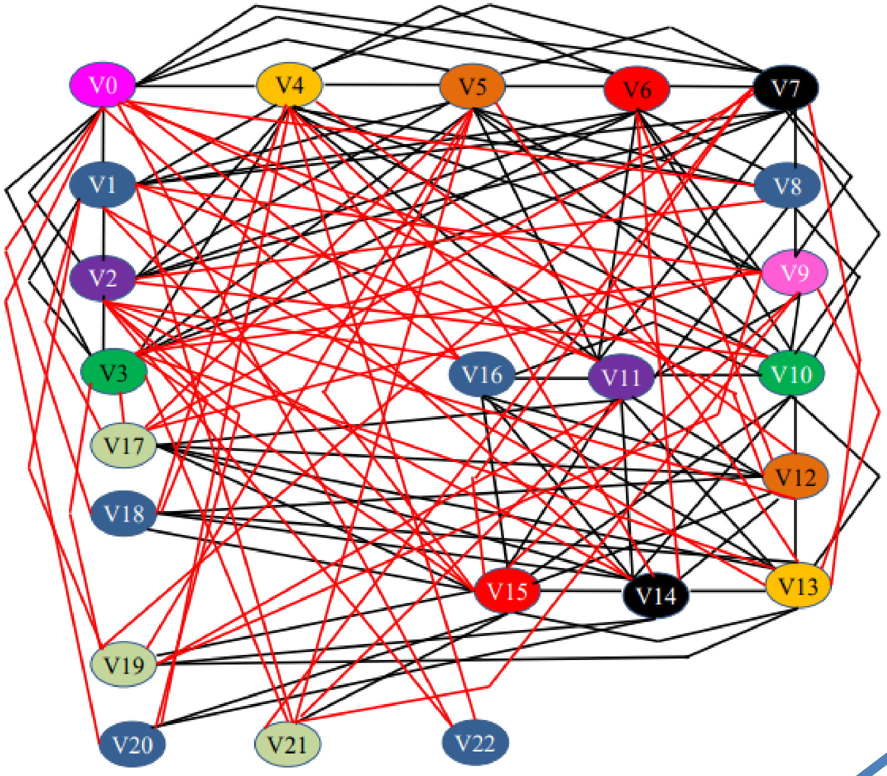
Original CIG

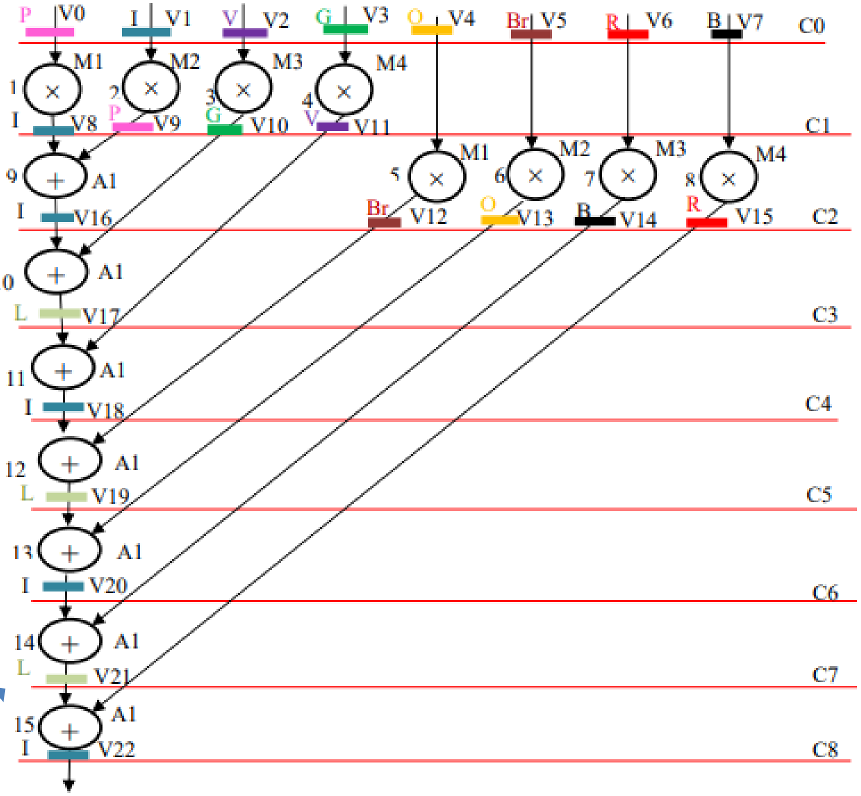


Changed registers to ensure no edges exist between registers in the same time span

CIG after embedding the above mentioned edges (shown in red)

# Modifying the SDFG to Accommodate the Newly Embedded Edges



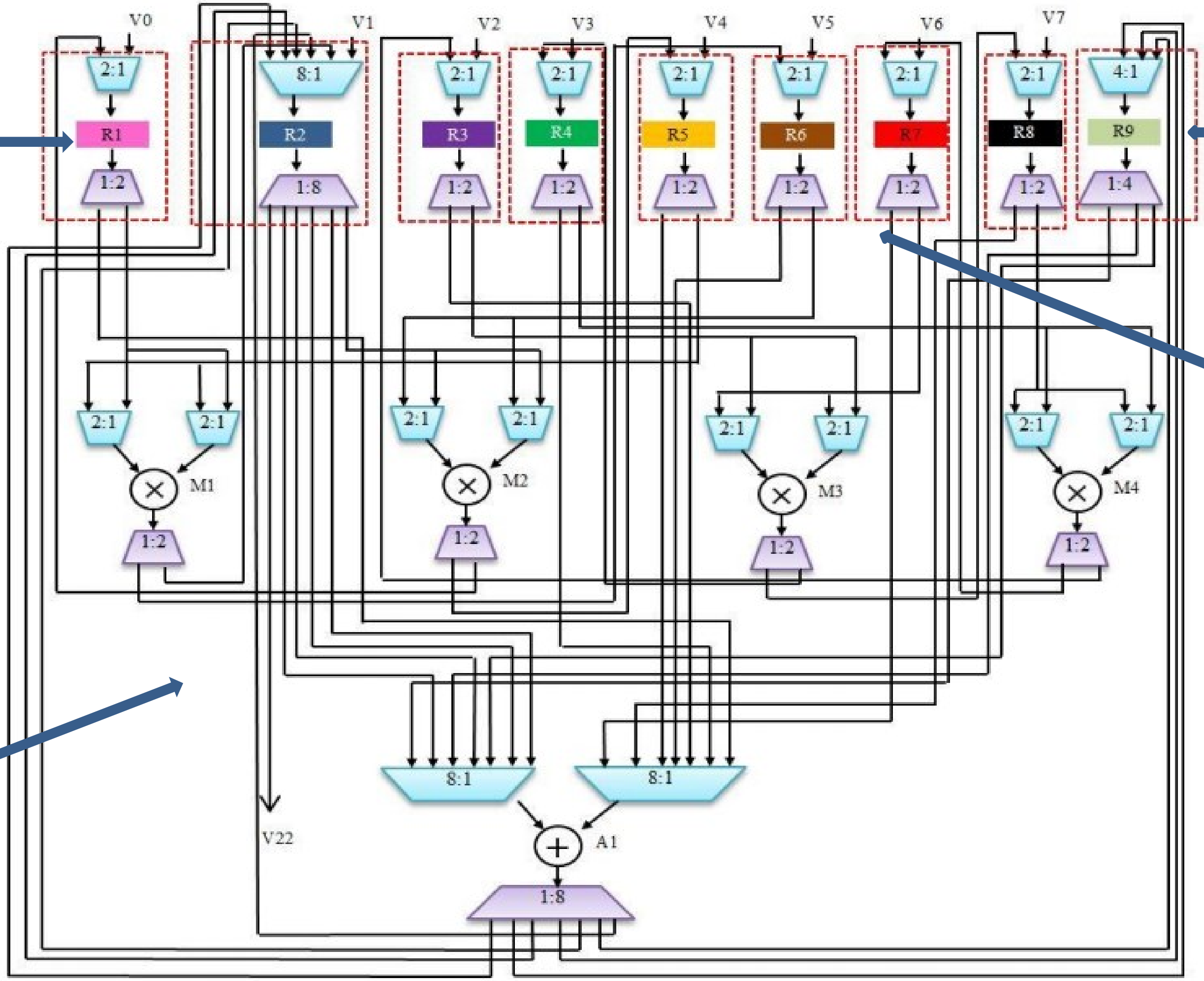


An extra register, 'Lime' (L) had to be incorporated and hence, an overhead of 1.

Scheduled Data Flow Graph constructed from the CIG after embedding secret hardware security constraints in the form of red edges.

# RTL Datapath of the Secured Design

R1, which previously had 9 inputs now has only 2



New register, R9 added as a result of securing the design.

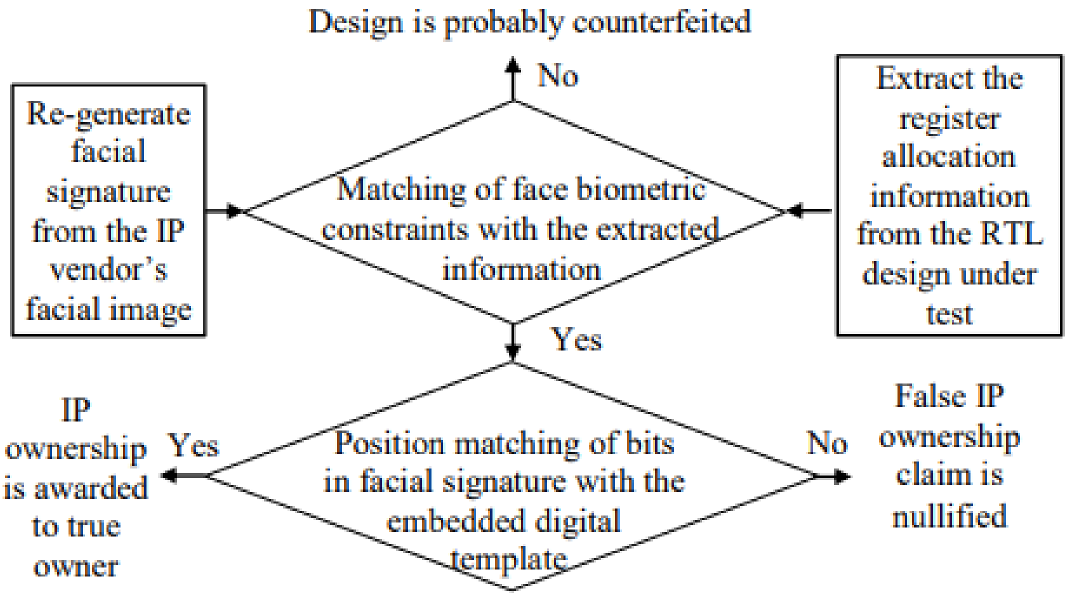
The red boxes show the changes in variables allotted to registers as a result of the embedded facial biometric. Notice how inconspicuous the hardware security constraints are.

Notice there is no change in the number of functional units.

RESOURCES IN THE RTL DATAPATH OF 8-POINT DCT (PRE AND POST EMBEDDING FACIAL BIOMETRIC CONSTRAINTS

| Resources pre-embedding security constraints | | | | Resources post-embedding security constraints | | | |
|---|---|---|---|---|---|---|---|
| FUs | # of registers | Multiplexers | Demultiplexers | FUs | # of registers | Multiplexers | Demultiplexers |
| 4M<br>1A | 8 | # 16x1 Muxes=1<br># 8x1 Muxes=2<br># 2x1 Muxes =15 | # 1x16 Demuxes=1<br># 1x8 Demuxes=1<br># 1x2 Demuxes=11 | 4M<br>1A | 9 | # 8x1 Muxes =3<br># 4x1 Muxes =1<br># 2x1 Muxes =15 | # 1x8 Demuxes=2<br># 1x4 Demuxes=1<br># 1x2 Demuxes=11 |

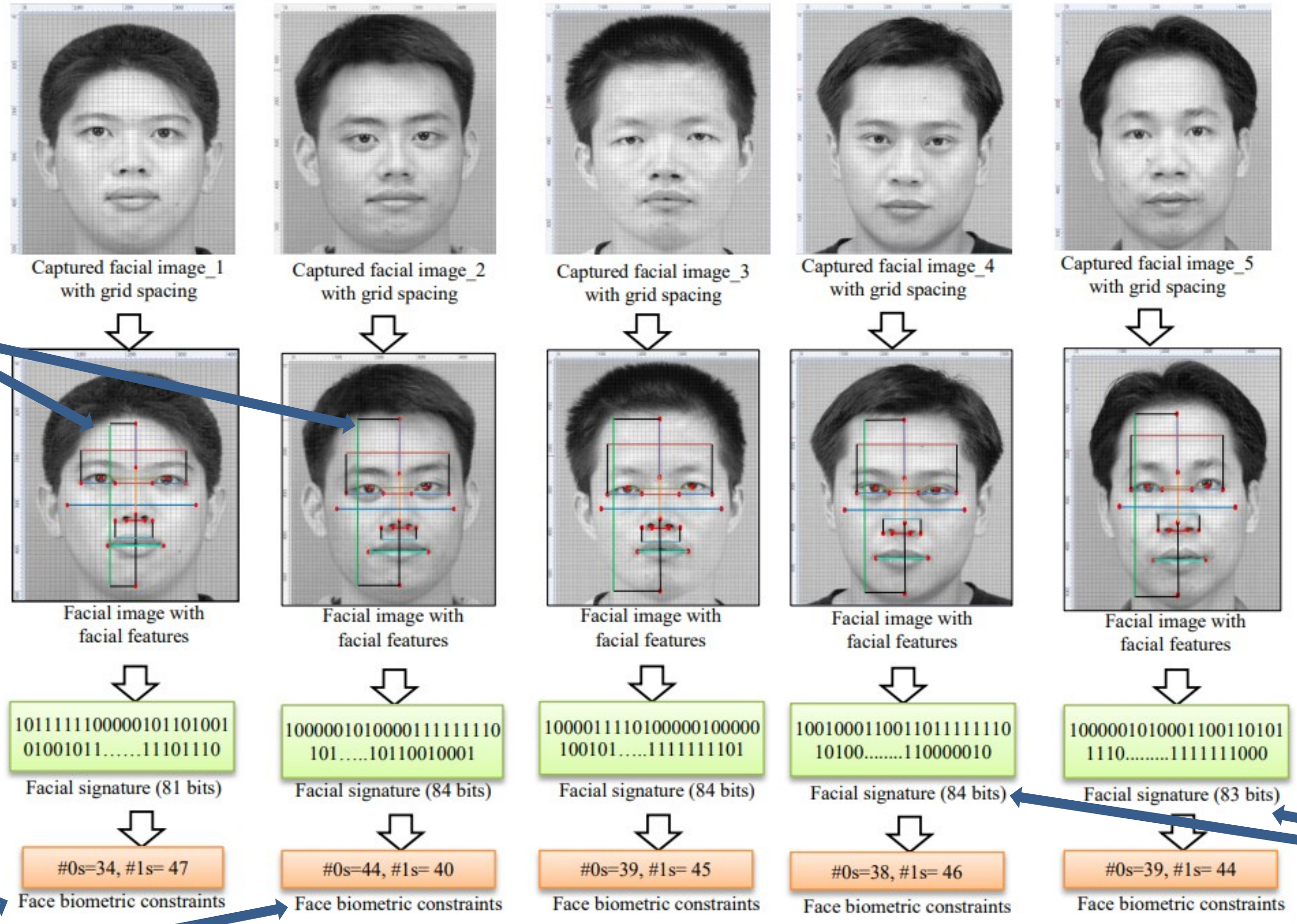
# Detection of Face Biometric Embedded into Design


Design is probably counterfeited
No
Re-generate facial signature from the IP vendor's facial image
Matching of face biometric constraints with the extracted information
Extract the register allocation information from the RTL design under test
Yes
IP ownership is awarded to true owner
Yes
Position matching of bits in facial signature with the embedded digital template
No
False IP ownership claim is nullified

*Too many details to be known to the adversary*:

- Precise coordinates of the nodal points
- Type of features selected
- Ordering of the features
- Number of 0s and 1s
- Positions of 0s and 1s
- Grid size and spacing

# Impact of Varying Facial Signature on Generated Hardware Security Constraints


Captured facial image_1 with grid spacing
Captured facial image_2 with grid spacing
Captured facial image_3 with grid spacing
Captured facial image_4 with grid spacing
Captured facial image_5 with grid spacing
Facial image with facial features
Facial image with facial features
Facial image with facial features
Facial image with facial features
Facial image with facial features
101111111000000101101001 01001011……11101110
Facial signature (81 bits)
1000001010000111111110 101…..10110010001
Facial signature (84 bits)
10000111101000000100000 100101…..1111111101
Facial signature (84 bits)
1001000110011011111110 10100........110000010
Facial signature (84 bits)
100000101000110011010 1 1110.........1111111000
Facial signature (83 bits)
#0s=34, #1s= 47
Face biometric constraints
#0s=44, #1s= 40
Face biometric constraints
#0s=39, #1s= 45
Face biometric constraints
#0s=38, #1s= 46
Face biometric constraints
#0s=39, #1s= 44
Face biometric constraints

Notice the varying feature dimensions.

Notice the varying length of digital templates.

Notice the varying number of 0s and 1s.

Digital Signature Templates for Facial Biometrics of 5 Different People

*More the number of features included, more likely the templates are bound to differ at atleast one bit position, thereby enhancing security.*

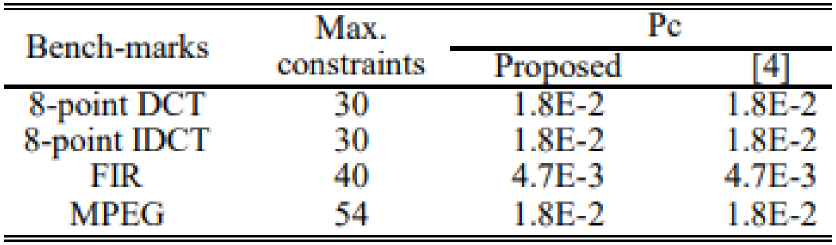
# Security Analysis

Probability of Coincidence, $P_c = \left(1 - \frac{1}{h}\right)^g$

h → number of registers in the CIG before implanting secret constraints

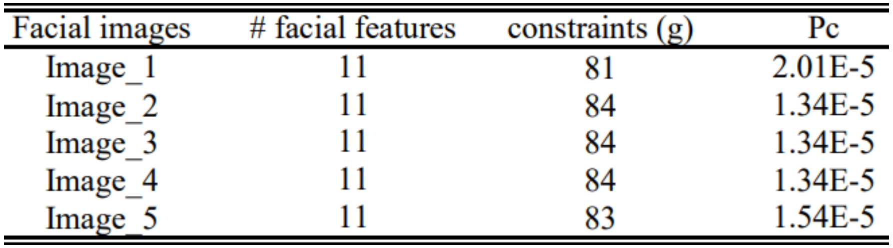
g → number of secret constraint edges added to the CIG

**Variation in $P_c$ for DCT for different facial images with same number of features**



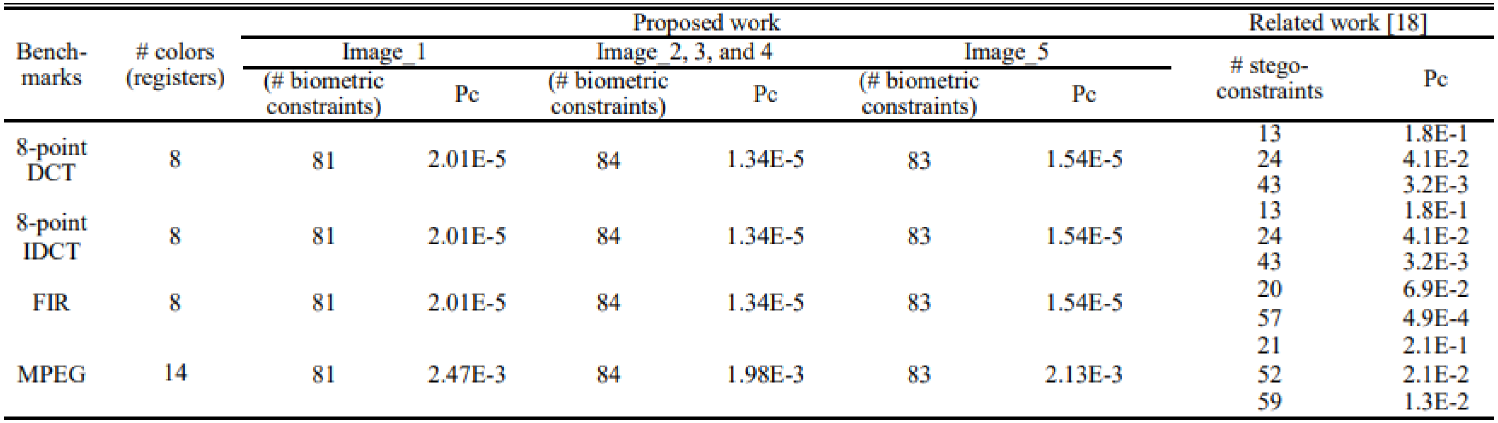
| Bench-marks | Max. constraints | Pc | |
|---|---|---|---|
| | | Proposed | [4] |
| 8-point DCT | 30 | 1.8E-2 | 1.8E-2 |
| 8-point IDCT | 30 | 1.8E-2 | 1.8E-2 |
| FIR | 40 | 4.7E-3 | 4.7E-3 |
| MPEG | 54 | 1.8E-2 | 1.8E-2 |

**Comparison of $P_c$ w.r.t to Fingerprint Biometric Approach [4]**



| Facial images | # facial features | constraints (g) | Pc |
|---|---|---|---|
| Image_1 | 11 | 81 | 2.01E-5 |
| Image_2 | 11 | 84 | 1.34E-5 |
| Image_3 | 11 | 84 | 1.34E-5 |
| Image_4 | 11 | 84 | 1.34E-5 |
| Image_5 | 11 | 83 | 1.54E-5 |

**Comparison of $P_c$ w.r.t Hardware Steganography [18] for the facial Image 1 to Image 5**



| Bench-marks | # colors (registers) | Proposed work | | | | | | Related work [18] | |
|---|---|---|---|---|---|---|---|---|---|
| | | Image_1 | | Image_2, 3, and 4 | | Image_5 | | # stego-constraints | Pc |
| | | (# biometric constraints) | Pc | (# biometric constraints) | Pc | (# biometric constraints) | Pc | | |
| 8-point DCT | 8 | 81 | 2.01E-5 | 84 | 1.34E-5 | 83 | 1.54E-5 | 13<br>24<br>43 | 1.8E-1<br>4.1E-2<br>3.2E-3 |
| 8-point IDCT | 8 | 81 | 2.01E-5 | 84 | 1.34E-5 | 83 | 1.54E-5 | 13<br>24<br>43 | 1.8E-1<br>4.1E-2<br>3.2E-3 |
| FIR | 8 | 81 | 2.01E-5 | 84 | 1.34E-5 | 83 | 1.54E-5 | 20<br>57 | 6.9E-2<br>4.9E-4 |
| MPEG | 14 | 81 | 2.47E-3 | 84 | 1.98E-3 | 83 | 2.13E-3 | 21<br>52<br>59 | 2.1E-1<br>2.1E-2<br>1.3E-2 |

Notice how Pc decreases as number of constraints increase.

*Note: The probability of coincidence represents the probability of coincidently detecting hardware security constraints in an unsecured design. Since we don't want any coincidence between signatures of original IP vendor and adversary, it is desirable for Pc to be as low as possible.*

# Design Cost Analysis

Design Cost, $C_f(Z_i) = \omega_1 \frac{L_d}{L_m} + \omega_2 \frac{A_d}{A_m}$

$Z_i$ → resource constraints

$A_d$ and $L_d$ → design area and latency respectively

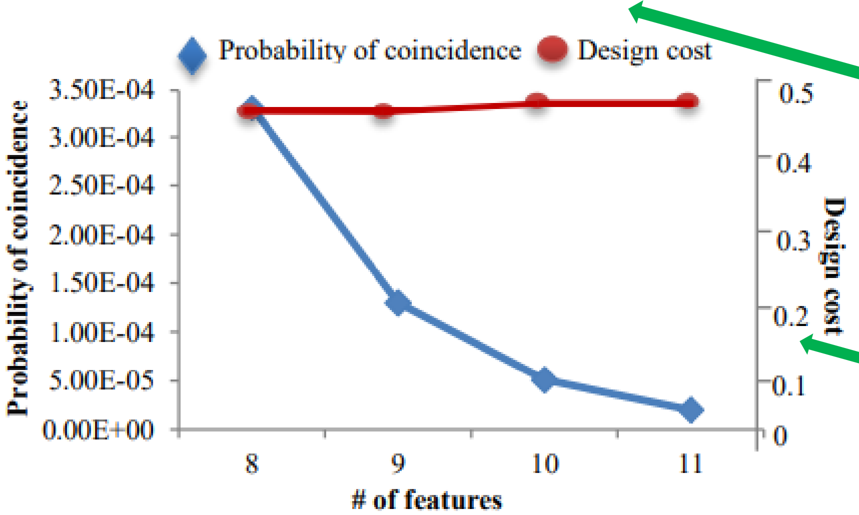
$A_m$ and $L_m$ → maximum area and latency of the design respectively

ω1 and ω2 → weights of latency and area in the cost function

**Design Cost Comparison of Facial Biometric Approach [4] with Fingerprint Biometric Approach and Baseline**

| Benchmarks | Design cost of pre-embedded design (baseline) | Design cost of fingerprint biometric approach [4] | Design cost of proposed facial biometric approach |
|---|---|---|---|
| 8-point DCT | 0.4721 | 0.5327 | 0.4727 |
| 8-point IDCT | 0.4721 | 0. 5327 | 0.4727 |
| FIR | 0.4443 | 0.4939 | 0.4443 |
| MPEG | 0.37046 | 0.37080 | 0.37080 |

*A 15nm NanGate library is used to calculate the latency and area of the design.*



Notice that the design cost is comparable to the baseline as there is almost no design overhead involved.

Notice how Design Cost remains almost constant while $P_c$ drops exponentially.

# Our CAD tools

Some CAD tools designed by us and hosted @ University of Florida – Cybersecurity Center, USA:

https://cadforassurance.org/tools/ip-ic-protection/crypto-steganography-tool/

https://cadforassurance.org/tools/ip-ic-protection/khc-stego/

# References

- Anirban Sengupta "Frontiers in Securing IP Cores - Forensic detective control and obfuscation techniques", The Institute of Engineering and Technology (IET), 2020, ISBN-10: 1-83953-031-6, ISBN-13: 978-1-83953-031-9

- Anirban Sengupta, Mahendra Rathor "Protecting DSP Kernels using Robust Hologram based Obfuscation", IEEE Transactions on Consumer Electronics, Volume: 65, Issue: 1, Feb 2019, pp. 99-108

- Anirban Sengupta, Saraju P. Mohanty "IP Core Protection and Hardware-Assisted Security for Consumer Electronics", The Institute of Engineering and Technology (IET), 2019, Book ISBN: 978-1-78561-799-7, e-ISBN: 978-1-78561-800-0

- Anirban Sengupta, Dipanjan Roy, Saraju P Mohanty, "Triple-Phase Watermarking for Reusable IP Core Protection during Architecture Synthesis", IEEE Transactions on Computer Aided Design of Integrated Circuits & Systems (TCAD), Volume: 37, Issue: 4, April 2018, pp. 742 – 755

- Anirban Sengupta, Mahendra Rathor, "IP Core Steganography using Switch based Key-driven Hash-chaining and Encoding for Securing DSP kernels used in CE Systems", IEEE Transactions on Consumer Electronics (TCE), 2020

- Anirban Sengupta, Mahendra Rathor "IP Core Steganography for Protecting DSP Kernels used in CE Systems", IEEE Transactions on Consumer Electronics (TCE) , Volume: 65 , Issue: 4 , Nov. 2019, pp. 506 – 515

- https://cadforassurance.org/tools/ip-ic-protection/faciometric-hardware-security-tool

- Anirban Sengupta, Rahul Chaurasia, Tarun Reddy "Contact-less Palmprint Biometric for Securing DSP Coprocessors used in CE systems", IEEE Transactions on Consumer Electronics (TCE) , Volume: 67, Issue: 3, August 2021, pp. 202-213

- Rahul Chaurasia, Aditya Anshul, Anirban Sengupta "Palmprint Biometric vs Encrypted Hash based Digital Signature for Securing DSP Cores Used in CE systems", IEEE Consumer Electronics (CEM) , Volume: 11, Issue: 5, September 2022, pp. 73-80

- Anirban Sengupta, Deepak Kachave, Dipanjan Roy "Low Cost Functional Obfuscation of Reusable IP Cores used in CE Hardware through Robust Locking", IEEE Transactions on Computer Aided Design of Integrated Circuits & Systems (TCAD), Volume: 38, Issue 4, April 2019, pp. 604 - 616

- Anirban Sengupta, Mahendra Rathor "Securing Hardware Accelerators for CE Systems using Biometric Fingerprinting", IEEE Transactions on Very Large Scale Integration Systems , Vol 28, Issue: 9, 2020, pp. 1979-1992

- Anirban Sengupta, E. Ranjith Kumar, N. Prajwal Chandra "Embedding Digital Signature using Encrypted-Hashing for Protection of DSP cores in CE", IEEE Transactions on Consumer Electronics (TCE), Volume: 65, Issue:3, Aug 2019, pp. 398 – 407

- Anirban Sengupta, Saumya Bhadauria, "Exploring Low Cost Optimal Watermark for Reusable IP Cores during High Level Synthesis", IEEE Access, Invited paper, Volume:4, Issue: 99, pp. 2198 - 2215, May 2016 .

# Conclusion

The future of VLSI is Energy-Security Tradeoff..

**Thank you**