SWIFT: Swarm Intelligence Driven ESL Synthesis for Functional Trojan Fortification

Anirban Sengupta[1], Rahul Chaurasia[2]

[1]Dept. of CSE, Indian Institute of Technology (IIT) Indore, India
[2]Dept. of CSE, Indian Institute of InformationTechnology(IIIT)Bhopal,India

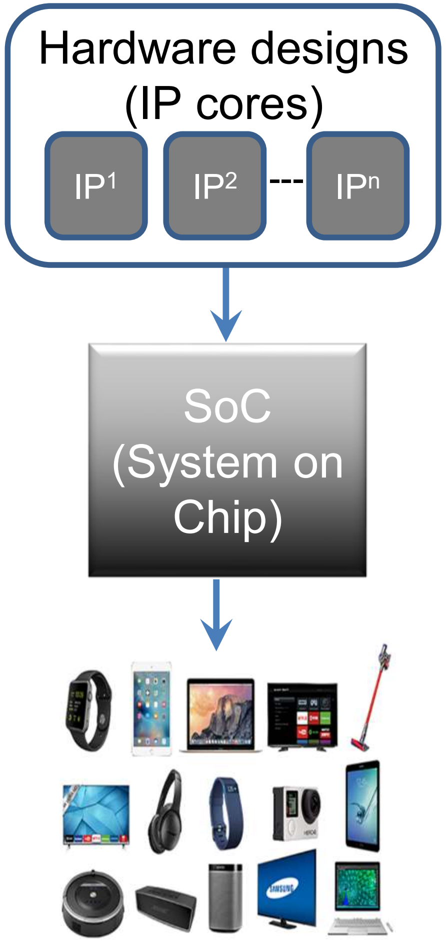
#ISES2024

Outline

- Introduction and Motivation
- Threat Model
- Prior Research
- Proposed Solution
- Methodology
- Experimental Setup and Results Analysis
- Conclusion

Introduction and Motivation

- Reusable hardware designs form key components of SoCs used in computing/CE systems.
- **Globalized supply chain** involves **untrustworthy third-party IP (3PIP) vendor** houses.
- Wide usability of DSP hardware designs/cores.
- Backdoor functional Trojans are a major concern for ensuring **reliable functionality** of CE systems and also for ensuring **consumer trust**.
- **Why it is crucial to address security threats** from the IP vendor level to the system integration level?
- **Detection and isolation** of such counterfeited hardware designs is essential as they may contain malicious Trojan.
- The threat of **hardware Trojan** is highly severe as it **can be implanted at any stage** of the IP design chain.
- For a CE system's secure and reliable operation, the integrated hardware designs must be Trojan fortified.



SoCs: system-on-chips, **CE**: Consumer Electronics

Threat Model

- Hardware designs or micro-IPs from untrustworthy IP core vendors (generally available in the module library of ESL synthesis tools) are susceptible to severe hardware threats.

- Further, the integration of 3PIPs from untrustworthy entities during in-house system design integration (System on chip (SoC) integration) may result in malfunctioning of CE systems and a security threat to end consumers.

- Hardware Trojan can be implanted secretly at any stage of the design chain by an adversary, which might not get detected during the Trojan detection process.

❖ This paper handles such functional Trojan, secretly embedded in the reusable DSP hardware designs (from 3PIP vendors) used in CE systems.

Prior Research

- In **[5]**, a scheme involving approximation circuits is introduced to hinder hardware Trojan inclusion at the gate level, showcasing its efficacy on gate-level benchmarks.

○ However, **[5]** does not address functional hardware Trojan quarantine in DSP hardware, particularly in the context of 3PIP designs.

○ Furthermore, swarm intelligence based exploration for optimizing secure Trojan-fortified architecture is not integrated into **[5]**.

- In **[6]**, authors propose DMR logic for Trojan detection in DSP cores.

○ However, it lacks capability in providing Trojan isolation and fortification.

- In **[7]**, authors design hardware Trojan-infected adversaries using functional camouflaging, discussing ways to furtively introduce Trojans at sites with low centrality magnitudes.

○ However, it does not address the design of low-cost, optimized Trojan-fortified DSP circuits.

❑ The proposed methodology ensures the generation of low-cost, Trojan-fortified designs using swarm intelligence based DSE, tailored to reusable DSP hardware utilized in the CE system.

Key Contributions/Proposed solution

❑ This work introduces several innovative contributions, including:

- A novel ESL synthesis-based approach for generating low-cost optimal Trojan fortified design architecture for securing DSP hardware designs against functional Trojan (causing erroneous functional output).

- Employing a distinct multivendor allocation policy based on Triple Modular Redundancy to ensure design fortification against functional Trojan.

- The proposed approach presents an integrated swarm intelligence based DSE framework with the Trojan fortification design methodology for exploring the optimal low-cost architectural solution.

Methodology - Design Flow of the Proposed Work

- ❑ The proposed approach comprises of two interdependent modules:

(i) Constructing the scheduled data flow graph (SDFG) of Trojan fortified TMR design and subsequently, based on the SDFG estimating the delay and area of the design and finally computing the fitness cost.

(ii) Performing the design space exploration.

✓ Ensuring a low-cost optimal Trojan fortified datapath design for reusable DSP hardware.

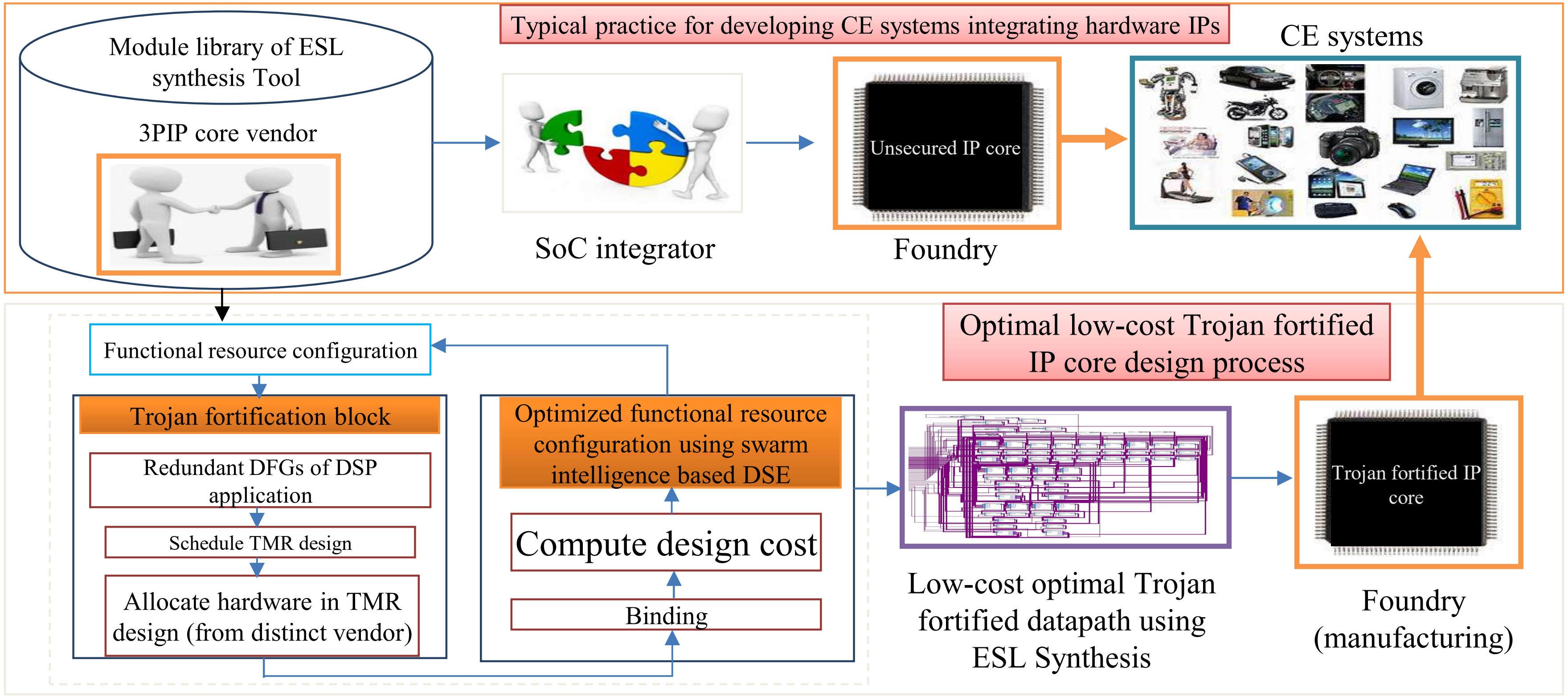
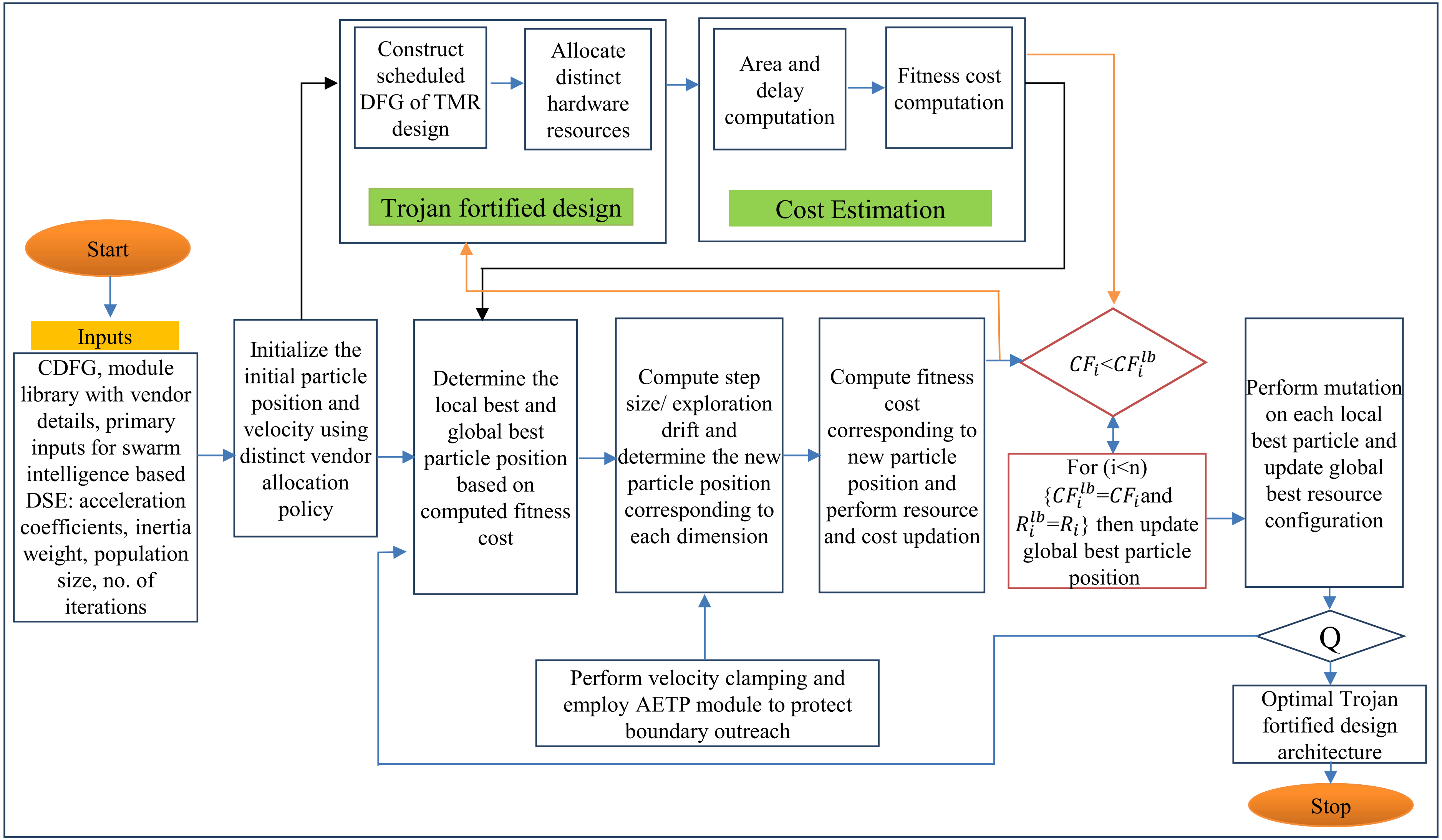

Typical practice for developing CE systems integrating hardware IPs
Module library of ESL synthesis Tool
3PIP core vendor
SoC integrator
Unsecured IP core
Foundry
CE systems
Functional resource configuration
Trojan fortification block
Redundant DFGs of DSP application
Schedule TMR design
Allocate hardware in TMR design (from distinct vendor)
Optimized functional resource configuration using swarm intelligence based DSE
Compute design cost
Binding
Optimal low-cost Trojan fortified IP core design process
Low-cost optimal Trojan fortified datapath using ESL Synthesis
Trojan fortified IP core
Foundry (manufacturing)

Fig. 1 Proposed Swarm Intelligence DSE based Trojan fortification process during ESL Synthesis (SWIFT)

Module-1: Proposed Trojan Fortification Design Block

- In this module, data flow graph (DFG) of the sample DSP design is constructed based on the initial functional resource configuration (particle position).
- Next, the design TMR logic, by combining two sister units (duplicate unit and triplicate unit) are built by duplicating the operations of the primary/original DFG (first unit) conforming to the DSP design.
- This combined DFG is scheduled using the LIST scheduling technique.
- A distinct multi-vendor allocation policy leverages the design with easy isolation of Trojan infected unit compared to allocating the resources from multiple vendors to a single unit.
- The information of the multiple vendors used during the allocation of the TMR-DSP hardware is confidentially known only to the system integrator.
- Likelihood of collusion between divergent unknown 3PIP vendors to realise the same Trojan payload is extremely little.
- A fault-tolerant voter then passes the functionally correct output.

Module-2: Integrating Swarm Intelligence based DSE



Start

Inputs

CDFG, module library with vendor details, primary inputs for swarm intelligence based DSE: acceleration coefficients, inertia weight, population size, no. of iterations

Initialize the initial particle position and velocity using distinct vendor allocation policy

Construct scheduled DFG of TMR design

Allocate distinct hardware resources

Trojan fortified design

Area and delay computation

Fitness cost computation

Cost Estimation

Determine the local best and global best particle position based on computed fitness cost

Compute step size/ exploration drift and determine the new particle position corresponding to each dimension

Perform velocity clamping and employ AETP module to protect boundary outreach

Compute fitness cost corresponding to new particle position and perform resource and cost updation

$CF_i<CF_i^{lb}$

For (i<n) $\{CF_i^{lb}=CF_i$and $R_i^{lb}=R_i\}$ then update global best particle position

Perform mutation on each local best particle and update global best resource configuration

Q

Optimal Trojan fortified design architecture

Stop

Fig.2. Proposed Swarm Intelligence framework in the Trojan fortification design methodology (SWIFT)

#ISES2024

Module-3: Demonstration of the Proposed Approach: SWIFT

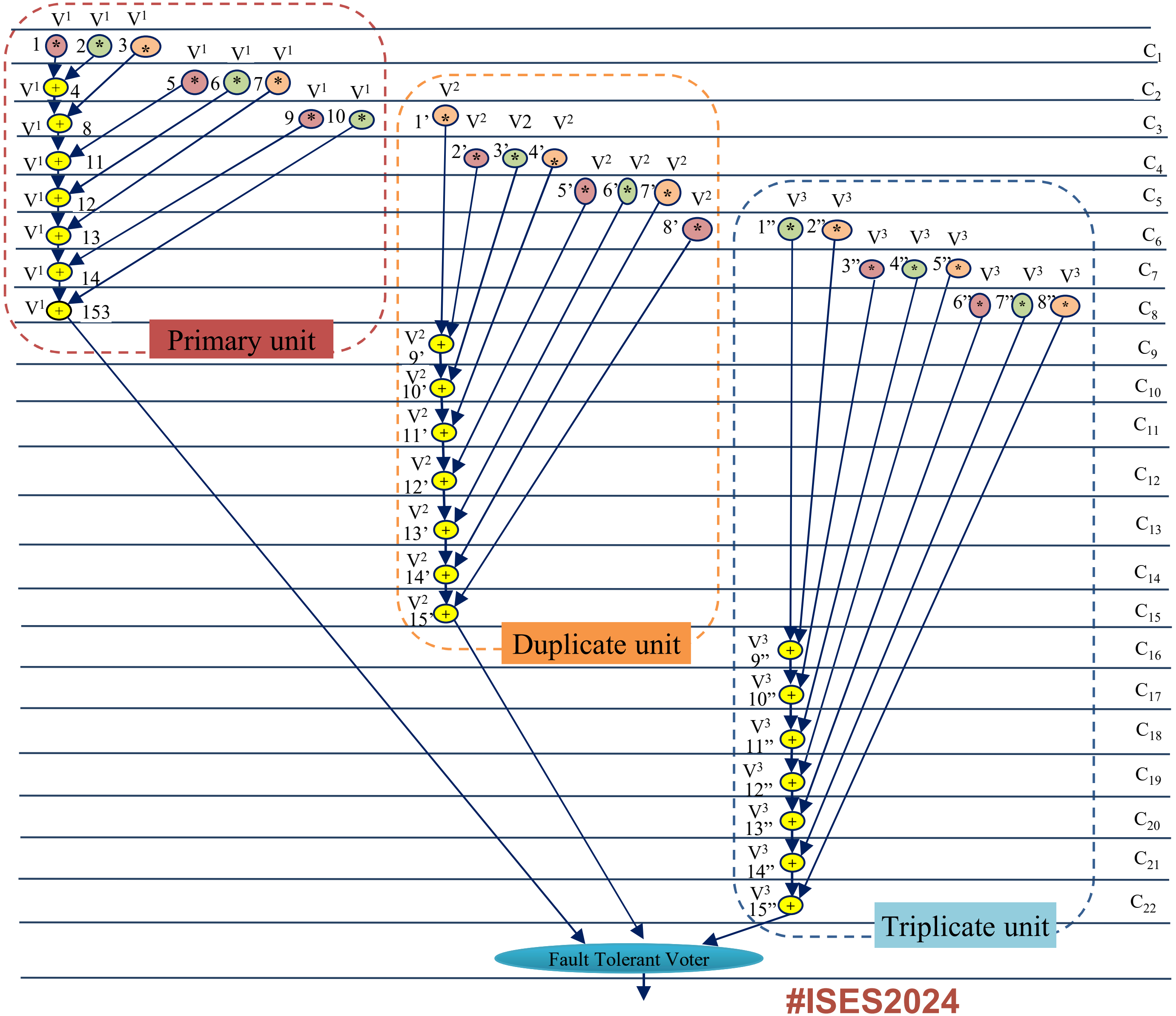


Fig. 3. Demonstration of Trojan fortified 8-point DCT core with distinct multivendor policy

#ISES2024

Experimental Setup and Results Analysis

- In the proposed security approach, parameters such as input sample DSP application, multi-vendor resources, DSE metrics such as inertia weight (linearly decreasing between 0.9 to 0.1), population size (number of particles=3, 5), acceleration coefficients and the total number of iterations, are considered during experimental evaluation and a 2.30 GHz processor and 4 GB of RAM is used for implementation.

- The proposed methodology is analysed in terms of ***exploration time*** and ***design cost overhead*** (for Trojan fortified DSP IP core), and ***security*** (the number of vulnerabilities handled).

A. Design Cost Function

- The fitness function includes normalized area and execution time corresponding to the architectural design of Trojan resistant TMR schedule and can be formulated as follows:

$$Fitness\ \text{cost} = W_1 * \left(\frac{A_{TMR}}{A_{MAX}}\right) + W_2 * \left(\frac{T_{TMR}}{T_{MAX}}\right) \qquad (3)$$

Here W_1 and W_2 are designer-defined weighing factors (W_1=W_2=0.5) to provide equal weightage during cost function evaluation. Further, A_{MAX} and T_{MAX} represents maximum design area and delay while and represents the computed area and delay of the proposed Trojan resistant TMR DSP design.

Results Analysis and Comparison

- The exploration time (time consumed by the swarm intelligence based DSE process for exploring a global optimal Trojan fortified DSP architectural solution) has been reported in **Fig. 4**.
- The cost comparison of designing Trojan fortified hardware design using the proposed approach and Trojan detection approach [6] have been presented in **Fig. 5**.

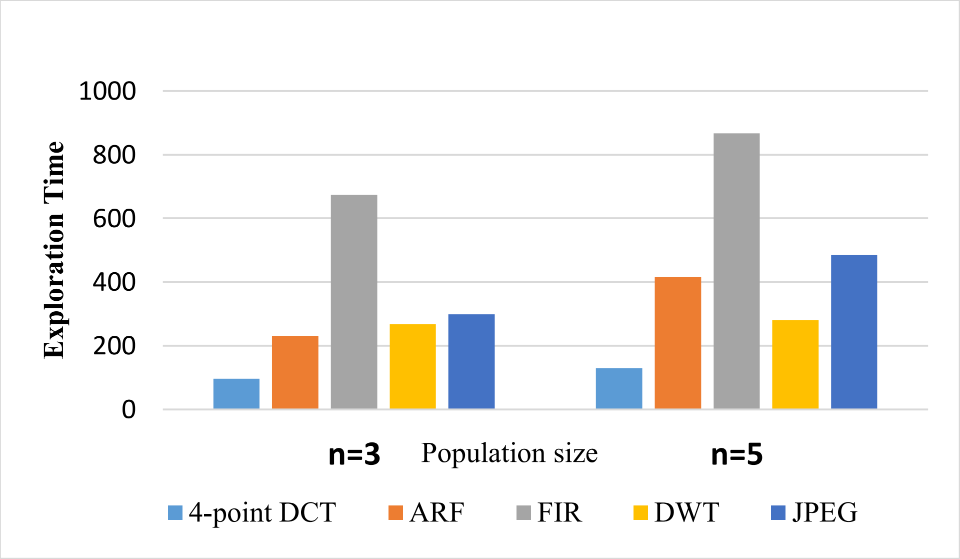


Fig. 4. Comparison of exploration time for producing Trojan fortified designs w r to varying population size

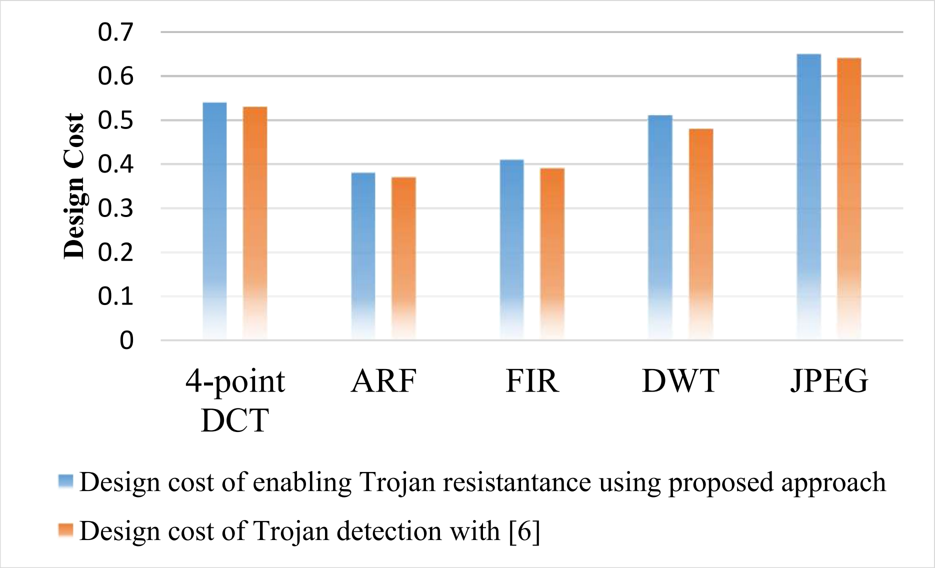


Fig. 5. Cost comparison of proposed approach with [6]

Cont.

- ❑ The proposed approach is equipped with generating the Trojan fortified design with minimal design cost overhead as evident from **Fig. 6**.
- ❑ The performance (delay) overhead of the proposed Trojan fortified TMR DSP designs compared to non-resistant designs, is shown in **Fig. 7**.

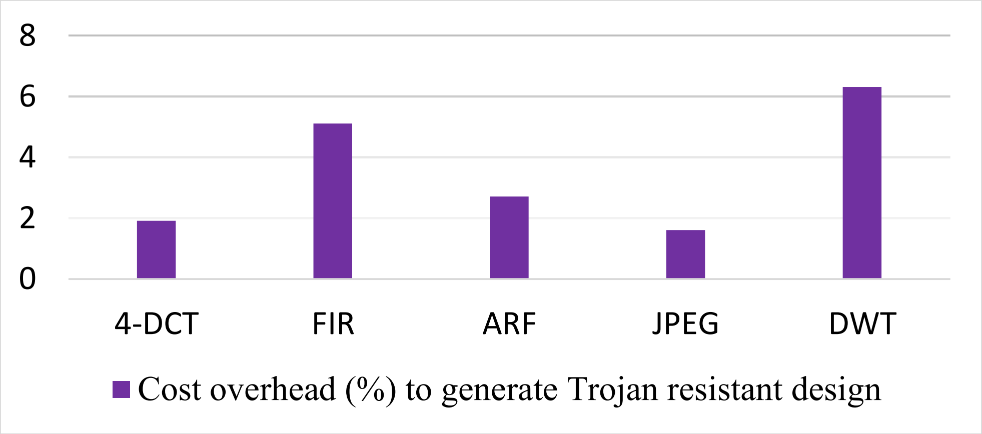


Fig. 6. % Cost overhead to generate Trojan fortified design

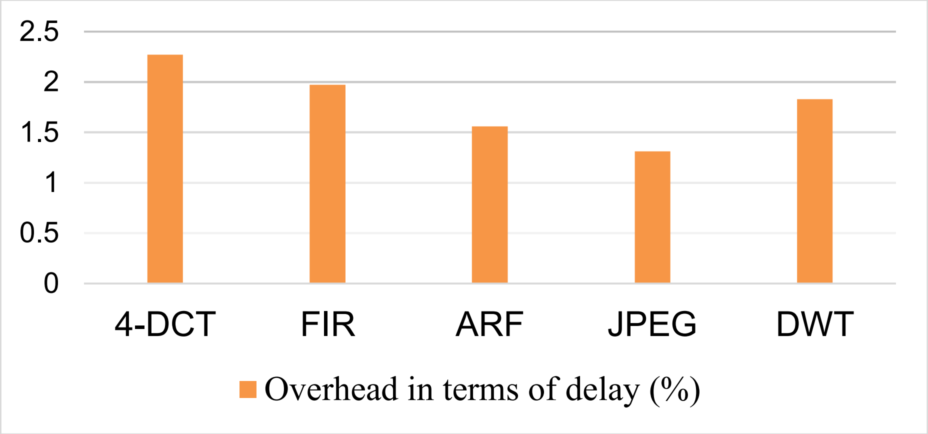


Fig. 7. % Delay overhead to generate Trojan fortified design (Average overhead is 1.79%)

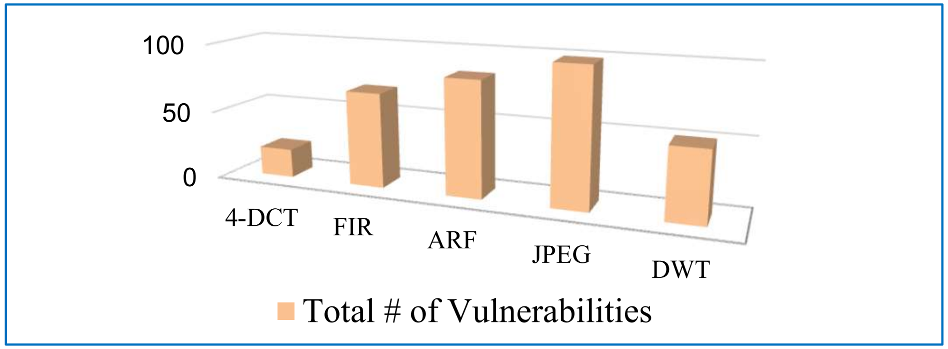


Fig.8. # of potential untrustworthy vulnerability sites tackled using proposed approach

#ISES2024

References

1. R. Chaurasia and A. Sengupta, “Security Vs Design Cost of Signature Driven Security Methodologies for Reusable Hardware IP Core,” *2022 IEEE International Symposium on Smart Electronic Systems (iSES)*, Warangal, India, 2022, pp. 283-288, doi: 10.1109/iSES54909.2022.00064.

2. F. Kounelis, N. Sklavos and P. Kitsos, “Run-Time Effect by Inserting Hardware Trojans, in Combinational Circuits,” *2017 Euromicro Conference on Digital System Design (DSD)*, 2017, pp. 287-290.

3. W. Hu, C. -H. Chang, A. Sengupta, S. Bhunia, R. Kastner and H. Li, “An Overview of Hardware Security and Trust: Threats, Countermeasures, and Design Tools,” *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 40, no. 6, pp. 1010-1038, 2021.

4. V. K. Mishra, A. Sengupta, “MO-PSE: Adaptive Multi Objective Particle Swarm Optimization Based Design Space Exploration in Architectural Synthesis for Application Specific Processor Design,” *Elsevier Journal on Advances in Engineering Software*, Volume 67, January 2014, pp. 111124.

5. H. Martin, L. Entrena, S. Dupuis and G. Di Natale, “A Novel Use of Approximate Circuits to Thwart Hardware Trojan Insertion and Provide Obfuscation,”*2018 IEEE 24th International Symposium on On-Line Testing And Robust System Design (IOLTS)*, 2018, pp. 41-42.

6. A. Sengupta, S. Bhadauria and S. P. Mohanty, “TL-HLS: Methodology for Low Cost Hardware Trojan Security Aware Scheduling With Optimal Loop Unrolling Factor During High Level Synthesis,” *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 36, no. 4,pp. 655-668, 2017, doi: 10.1109/TCAD.2016.2597232.

7. H. Li, A. Abdelhadi, R. Shi, J. Zhang and Q. Liu, “Adversarial Hardware With Functional and Topological Camouflage,” *IEEE Trans. Circuits Syst. II: Exp. Briefs*, vol. 68, no. 5, pp. 1685-1689, 2021, doi: 10.1109/TCSII.2021.3065292.

8. S. Salivahanan and A. Vallavaraj, “Digital Signal Processing”, McGraw-Hill Education (India) PvtLimited, 2001, ISBN: 9780074639962.

9. 15 nm open cell library. [Online], Available: https://si2.org/open-cell-library/, last accessed on Aug. 2023.

Conclusion

- This paper introduced a robust ESL synthesis based optimal low-cost Trojan fortification methodology using swarm intelligence based DSE called ‘SWIFT’.
- The proposed approach is capable to detect all functional Trojan at minimal design cost overhead.

Thank You