An Improved Publicly Detectable Watermarking Scheme Based on Scan Chain Ordering

Aijiao Cui and Chip-Hong Chang

Centre for High Performance Embedded Systems, Nanyang Technological University,
50 Nanyang Drive, Research Techno Plaza, 3rd Storey, Border X Block, Singapore 637553

***Abstract*— This paper proposes an improved version of watermarking scheme at the Design-for-Testability (DfT) stage for VLSI Intellectual Property (IP) Protection. The improved scheme overcomes the weaknesses of previous scan chain watermarking scheme by imposing the extra ordering constraints generated by the IP owner's signature on all scan flip-flops impartially. IP authorship can be publicly authenticated in the field by injecting a given test vector and matching a permuted output response vector against a transformed reference pattern. Both the output response and the reference sequence are related to a pseudorandom sequence generated by a public-key cryptographic algorithm. Experimental results show that the improved method has a low probability of coincidence and low test power overhead.**

I. INTRODUCTION

The System-on-Chip era is marked by the wide adoption of reusable IP cores. Trading of IP cores in an open environment poises new problem in tracking illegal IP distribution. Watermarking techniques [1]-[4] have been proposed as an active approach to protect the copyrights of VLSI designs. Based on the watermark detection, these techniques can be classified under static or dynamic watermarking schemes [5]. The dynamic watermarking [3], [4] avoids the risk of exposing the constraint generator in static watermarking [1], [2] during verification by allowing the authorship to be detected by running the watermarked IP with some specific input patterns. If the dynamic watermarking scheme is applied at the DfT stage [3], [4], the ownership can be directly detected in the field even after the IP core has been integrated into SoC and packaged.

In [3], we proposed a dynamic watermarking scheme based on the power-driven scan chain ordering problem. When a specific input vector is injected, the output response will contain the watermark at some selected scan flip-flop positions. This scheme is found to possess some weaknesses. To verify the authorship, the positions of these specific flip-flops must be known, which needs special measure to ensure its security for public authentication. Also, the test patterns distributed with the watermarked IP may create a security leak due to the biased selection of the watermarked flip-flops based on the weighted transitions of the test and response vectors. These make the watermark vulnerable to attack without removing the entire scan chain. These weaknesses arise because not all flip-flops in the scan chain have equal privileges to be allocated to every position specified by the extra constraints imposed by the watermark.

In this paper, we propose an improvement to overcome the above weaknesses in our DfT watermarking scheme [3]. Both the switching power minimization criterion and the extra constraints generated by the watermark information are considered for a more 'random' allocation of all flip-flops in the scan chain instead of giving some flip-flops the prerogative to assumed positions dictated by the specific output response vector [3]. A public-key cryptosystem is also used to assure that the authorship can be authenticated publicly without divulging the watermark information.

II. REVIEW OF PREVIOUS WORK

This section reviews our earlier watermarking scheme based on scan chain ordering [3]. The example in [3], as shown in Fig. 1, is used for the illustration. At the DfT stage, the extra constraints are imposed on the ordering of the flip-flops so that the watermark bits, w_1, w_2 and w_3 will be extracted from some reordered flip-flop positions, mp_1, mp_2 and mp_3, which are randomly generated by the signature. The watermarking process finds a permutation, $\pi_{wm}(R)$ of the scan flip-flops that will minimize the test switching power. When a specific input test vector is applied onto the scan-in pin, S_{in}, the watermark bits will be detected at positions mp_1, mp_2 and mp_3 of its output response vector. The watermarked scan chain, $\pi_{wm}(R) = r_2\ r_1\ r_5\ r_7\ r_4\ r_6\ r_3$ with $mp_1 = p_4$, $mp_2 = p_2$ and $mp_3 = p_7$ is shown at the bottom of Fig. 1, where p_i is the i-th flip-flop position of the watermarked scan chain.

To verify the authorship, the IP owner must be able to show that the bits extracted from p_4, p_2 and p_7 match the bits w_1, w_2 and w_3. By revealing the positions hosting the watermark bits publicly for authentication, it is easier for the attacker to erase the watermark information on those flip-flops without removing the scan chain. The watermarked constraint is imposed by a power-driven scan chain ordering algorithm. Based on the number of weighted transitions [6], a heuristic Nearest Neighbor (NN) greedy algorithm is used to re-allocate flip-flops into the scan chain. For the positions that host the watermark bits, the flip-flops are constrained to be selected from a subset of the unallocated flip-flops according to the values of the watermark bits. For example, in Fig. 1, as p_2 is a watermarked position, the search for the nearest neighbor to the flip-flop in p_1 (which is r_2 in this case) is limited only to the scan flip-flops r_1, r_4 and r_5. However, from the weighted transitions of the test patterns, the nearest neighbor to r_2 is actually r_3, which is not a member of $\{r_1, r_4, r_5\}$. Thus, from the test patterns, if the flip-flop is not the real nearest neighbor to the preceding flip-flop in the watermarked scan chain, the watermarked flip-flop and its information will be exposed, making the

978-1-4244-3828-0/09/$25.00 ©2009 IEEE

scheme vulnerable. These weaknesses will be overcome by the improved scheme proposed in the next section.

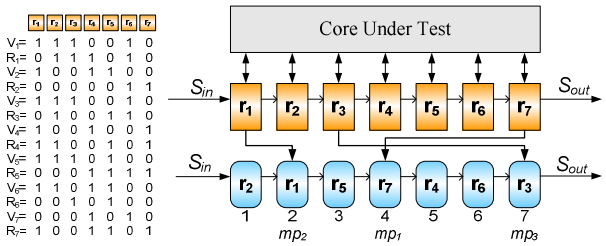

Fig. 1. Example of watermarking by scan cell ordering.

III. Improved Watermarking Scheme

Let $R=\{r_i\}_{i=1}^{N}$ be the original scan chain with N scan flip-flops and $P=\{p_j\}_{j=1}^{N}$ be the N flip-flop positions. A weighted connected graph, $G(V, E)$ can be built from the complete test set. Each vertex of the graph represents a scan flip-flop. An edge $E(r_i, r_j)$ connecting two flip-flops, r_i and r_j carries a weight equal to the total number of bit differences between r_i and r_j for the entire test set. The watermark is embedded to find a permutation $\pi_{wm}(R)$ with minimal test switching power such that a specific output vector can be obtained when a user-specific test vector is applied onto S_{in}. The watermarking process is detailed as follows:

A. *Watermark Generation*

The IP ownership is displayed by a meaningful text string M. This arbitrary length binary string is shortened to a fixed length hash-code, H using the secure hash algorithm, SHA-1 [7]. Directed by H, a pseudorandom number generator (PNG) [7] is used to generate a sequence of N unique numbers, $Q=\{q_i\}_{i=1}^{N}$, $1 \le q_i \le N$. This is the watermarked constraint used for the scan chain reordering.

B. *Generation of Reference Output Vector*

A designated input vector, $X=\{x_i\}_{i=1}^{N}$, $x_i \in \{0,1\}$, is shifted into the initial scan chain to obtain an output vector, $Y=\{y_i\}_{i=1}^{N}$, $y_i \in \{0,1\}$. Suppose there are N_0 "0" bits and N_1 "1" bits in Y. The N_0 and N_1 flip-flops are assigned to two sets, R_0 and R_1, respectively. Let Z be an N-bit binary vector where the first N_0 bits are "0" and the remaining N_1 bits are "1", i.e., $Z=\{z_i\}_{i=1}^{N}$, $z_i = 0$ for $1 \le i \le N_0$ and $z_i = 1$ for $(N_0+1) \le i \le N$. Q is encrypted using a private key, K_e, of a public-key cryptosystem (PKC) [7] to generate a random number sequence, $C=f_E(K_e,Q)=\{c_i\}_{i=1}^{N}$, $c_i = q_j$, $1 \le j \le N$. The bit sequence Z is then ordered according to C to generate $U=\{u_i\}_{i=1}^{N}$, where $u_i = z_{c_i}$.

C. *Waterarmk Insertion*

During the watermarking process, the allocation of a flip-flop to a position in the scan chain is directed by the integer sequence Q and the binary vector U. Let $p_j = \pi_{wm}(r_i)$ denote the mapping of a vertex, $r_i \in R$ to a position, $p_j \in P$. The first position, p_1 will be arbitrarily allocated to any flip-flop from R_ϕ, where $\phi \in \{0, 1\}$ and $\phi = u_{q_1}$. Then the allocated flip-flop, $\pi_{wm}^{-1}(p_1)$ is removed from R_ϕ. For all other positions, p_i, $2 \le i \le N$, the NN algorithm is used to select a flip-flop from R_ϕ, $\phi = u_{q_i}$, that has the least edge cost in the graph G to the vertex $\pi_{wm}^{-1}(p_{i-1})$. The selected flip-flop $\pi_{wm}^{-1}(p_i)$ is then removed from the set R_ϕ. This process is repeated until every scan flip-flop in R has been allocated to a unique position in the scan chain. The watermark insertion procedure is summarized by the pseudo code in Fig. 2.

watermark_insertion $(G, M, N, X, R, P, f_E, K_e)$ {

$H = \text{SHA-1}(M)$; $Q=\{q_i\}_{i=1}^{N} = \textbf{PNG}(H, N)$, $1 \le q_i \le N$;

Obtain $Y=\{y_i\}_{i=1}^{N}$ by applying $X=\{x_i\}_{i=1}^{N}$ onto R ;

$R_0 = \{r_i \in R \mid y_i = 0 \quad \forall 1 \le i \le N\}$, $N_0 = |R_0|$;

$R_1 = \{r_i \in R \mid y_i = 1 \quad \forall 1 \le i \le N\}$, $N_1 = |R_1|$;

$Z = \{z_i = 0 \mid \forall\ 1 \le i \le N_0, z_i = 1 \mid \forall\ (N_0+1) \le i \le N\}$;

$C = f_E(K_e, Q) = \{c_i\}_{i=1}^{N}$, $c_i = q_j$, $1 \le j \le N$;

$U = \{u_i\}_{i=1}^{N}$, $u_i = z_{c_i}$;

$\pi_{wm}^{-1}(p_1) \in R_\phi$, $u_{q_1} = \phi$, $\phi \in \{0,1\}$;

$R_\phi = R_\phi - \pi_{wm}^{-1}(p_1)$;

for (i = 2 **to** N) {

$(p_{i-1}, p_i) = \textbf{min}(E(\pi_{wm}^{-1}(p_{i-1}), r_j))$, $u_{q_i} = \phi$, $\phi \in \{0,1\}$, $r_j \in R_\phi$;

$R_\phi = R_\phi - \pi_{wm}^{-1}(p_i)$;

}

return π_{wm} ;

}

Fig. 2. Watermarking on power-driven scan chain ordering.

D. *Public and Field Verification of Authorship*

The verifier is given $X' = \pi_{wm}(X)$, the number sequence, C and the public key of the IP owner, K_d for authorship authentication. A response vector, Y' is obtained by loading X' onto the watermarked scan chain in the test mode. If $\hat{N}_0$ and $\hat{N}_1$ are numbers of "0" and "1" bits of Y', respectively, an N-bit binary vector, $\hat{Z}$ with $\hat{N}_0$ "0"s followed by $\hat{N}_1$ "1"s is created and permuted according to the order of C to obtain a reference sequence, $\hat{U}$. C is decrypted using the public key, K_d to obtain $Q = f_D(K_d, C)$. Q is then used to permute Y' to obtain a binary sequence U' such that the q_i-th ($q_i \in Q$) element of U' is equal to the i-th element of Y', i.e., $U'=\{u'_{q_i}\}_{i=1}^{N}=\{y'_i\}_{i=1}^{N}$. If U' perfectly matches or is highly correlated with $\hat{U}$, the authorship is verified as Q can only be generated by the IP owner.

E. *A Simple Example*

Consider the scan chain in Fig. 3. V_i and R_i on the left hand side denote the i-th test and response vectors, respectively for the original scan chain, $\pi(R) = r_1 r_2 r_3 r_4 r_5 r_6 r_7$. The connected graph for the scan chain is shown in Fig.

4(a). Suppose Y = "0110011" is obtained when X = "0010110" is loaded into the scan chain. From Y and $\pi(R)$, $R_0 = \{r_1, r_4, r_5\}$, $R_1 = \{r_2, r_3, r_6, r_7\}$ and Z = "0001111". The vertices corresponding to the flip-flops in R_0 and R_1 are marked with dot lines and continuous lines, respectively in Fig. 4. Assume that $Q = \{3, 4, 2, 7, 5, 1, 6\}$ and $C = f_E(K_e, Q) = \{4, 1, 6, 3, 5, 7, 2\}$. Then, according to Z and C, U = "1010110".

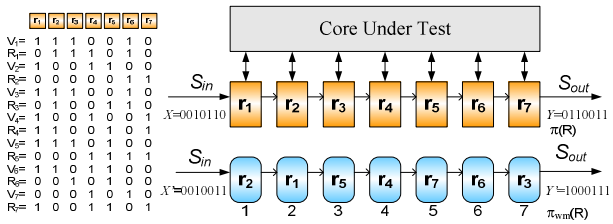

Fig. 3. Example of improved DfT watermarking scheme.

For $u_{q_1} = u_3 = 1$, the flip-flop allocated to p_1 must come from R_1 to produce a "1" bit in the first flip-flop of the watermarked scan chain. If r_2 is arbitrarily selected, R_1 is updated to $\{r_3, r_6, r_7\}$. As $u_{q_2} = u_4 = 0$, flip-flops from R_0 will be withdrawn to compete for p_2. From Fig. 4(b), the edge connecting r_1 to r_2 has the smallest cost among all the qualified edges connected to r_2, so r_1 is allocated to p_2 and removed from R_0. Similarly, r_5 is allocated to the next position, p_3 since $u_{q_3} = u_2 = 0$ and $e(r_1, r_5) < e(r_1, r_4)$. The process is repeated until all flip-flops are allocated. The ordering of the final watermarked scan chain is $\pi_{wm}(R) = r_2\, r_1\, r_5\, r_4\, r_7\, r_6\, r_3$. This is shown in the lower part of Fig. 3. To verify the IP authorship, the vector X' = "0010011" is loaded into the watermarked scan chain in the test mode to obtain Y' = "1000111", from which $\hat{Z}$ = "0001111" is produced. According to Y' and the given C, the sequence $\hat{Z}$ is permuted to obtain $\hat{U}$ = "1010110". C is then decrypted with the public key K_d to obtain $Q = \{3, 4, 2, 7, 5, 1, 6\}$. Y' is permuted according to Q, to obtain U'. If U' matches $\hat{U}$, the authorship is proved. Otherwise, the watermark is either not present or has been modified or erased.

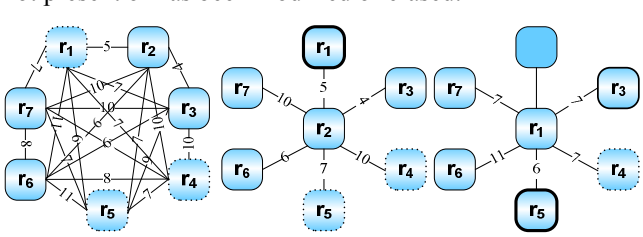
(a) cost graph (b) flip-flop for p_2 (c) flip-flop for p_3

Fig. 4. Watermarking process according to the cost graph.

IV. Watermark Resilience Analysis

The probability of coincidence, P_c is a key measure of the strength of a watermarking scheme. It denotes the probability that a non-watermarked design carries the watermark by coincidence. For our proposed scheme, if the output sequence under the input vector X' has the same permutation of "0" and "1" bits as Y', the design is said to possess the watermark. P_c can then be expressed as:

$$P_c = \frac{P_{N_0}^{N_0} \cdot P_{N_1}^{N_1}}{P_N^N} = \frac{N_0!\,N_1!}{N!} \tag{1}$$

A lower P_c implies a stronger authorship proof. From (1), the longer the scan chain length and the more balance between the numbers of "0" and "1" bits in Y', the stronger the watermarking scheme.

False positive response occurs when Y' is obtained from the injection of some input vectors other than X' into the watermarked scan chain. If there are $N_C(\tau)$ output responses that match at least τ ($0 \le \tau \le 1$) fraction of the bits of Y' when N_T randomly generated test vectors are applied onto the scan chain, then we can define:

$$P_\lambda(\tau) = \frac{N_C(\tau)}{N_T} \tag{2}$$

where $P_\lambda(\tau = 1)$ gives the exact false positive rate. When τ decreases, P_λ increases and a threshold for τ can be determined empirically that with high degree of confidence, Y' will not appear in the output responses for a given number of input vectors tested. Hence the authenticity of the design can be assured as long as $P_\lambda(\tau)$ is very low.

In what follows, four typical attacks are analyzed with Alice as the IP owner and Bob as the attacker. Variables with subscripts "A" and "B" are associated with Alice and Bob, respectively.

A. *Ghost Search*

Without touching Alice's design, Bob may load X_B' into Alice's watermarked design during test mode to obtain Y_B' and generate the sequence Z_B. He may refer to an encrypted number sequence, C_B to permute Z_B to generate U_B. Bob will then find a set of numbers, Q_B to realize the mapping of Y_B' to U_B. This attack scenario can not be substantiated as it is computationally infeasible for Bob to find a key to decrypt C_B to the sequence Q_B or reverse a keyed one-way PNG to generate Q_B. Alternatively, Bob may first generate Q_B using the PNG and then map the Y_B' to a bit sequence, U_B according to Q_B. However, it is computationally intractable for Bob to find a key pair (K_e, K_d) for the encryption of Q_B and the decryption of C_B [7].

B. *Denial Attacks*

Bob may deny that Alice has inserted her ownership information into the design. He may claim that Alice may have randomly found X_A' and the corresponding sets of Q_A, Z_A, C_A and U_A. To repudiate Bob's accusation, Alice needs to repeat the verification process in Section III.D and show that the probability for the output response to a randomly generated input vector other than X_A' to match Y_A' is very low, i.e., $P_\lambda(\tau = 1)$ must be very low.

C. *Removal Attacks*

It is obvious that no watermark can survive upon removal of the scan chain. The IP core of the scan chain is

assumed to be watermarked with other technique, for example, [2] before its test circuit is watermarked by the proposed scheme to enable public and field authentication. Without overly perturbing the useful test circuit, Bob may randomly reorder some scan flip-flops in the scan chain in order to alter the watermark information to invalidate the authorship proof during field authentication. Bob needs to change the circuit routing to reorder sufficient number of flip-flops to modify at least $(1-\tau)$ fraction of Y_A'. Assume α scan flip-flops can be reordered with reasonable effort and without obvious degradation to the circuit performance. Further assume that β ($\beta \leq \alpha$) bits will be altered in this process. In most cases, $\alpha < (1-\tau)\cdot N$ as N is usually much greater than α. This means that $\beta < (1-\tau)\cdot N$. It can be shown that it is difficult to alter enough bits to invalidate the authorship for a large N.

D. *Unauthorized Addition Attacks*

If Bob would like to insert his own information into Alice's design, he will have to remove the scan chain and repeat the proposed watermarking process to generate his own scan chain. This amounts to a task of difficulty equivalent to the complete repetition of the scan chain design without knowing the IP circuit.

V. EXPERIMENTAL RESULTS

In the experiment, the DfT tool, DFTadvisor and the ATPG tool, FastScan by Mentor Graphics are used to generate the initial vectors, Y from X on the originally optimized scan design. The watermarking scheme is applied on sequential circuits from ISCAS89, ISCAS99 and LGSynth93 benchmark suites. All calculations were performed on a 750–MHz Sun UltraSPARC-III with Solaris operating system and 2 GB of memory.

Tables I shows the results of watermarking 14 benchmark circuits. The set of pseudorandom numbers is generated according to the scan chain length, N. WT_{org} and WT_{wm} denote the number of weighted transitions of the originally optimized scan chain and that of the watermarked scan chain, respectively. ΔWT is calculated as the percentage increments from WT_{org} to WT_{wm}. A negative percentage implies performance improvement due to watermarking. The switching power overhead is less than 4% for all designs. The average overhead is less than 2%, which shows that the proposed watermarking scheme incurs very low overhead on the test power.

The watermark strength is evaluated by P_c in Table I. The value of P_c decreases rapidly as the scan chain length, N increases. When $N > 1000$, P_c reduced to less than 1.0×10^{-300}. To determine the false positive rate, P_λ, 200 randomly generated input vectors were applied on each watermarked design. None of the output responses was detected to perfectly match the known Y'. When τ is reduced to 0.6, P_λ remains zero for all designs. When τ is reduced to 0.5, all the watermarked designs have $P_\lambda \geq 50\%$. So it is reasonable to assume that when more than 40% of the bits are corrupted, the watermark can not be detected. If we assumed that $\alpha = 150$ flip-flops in the scan chain can be moved around with reasonable effort and without significantly degrading the circuit performance. The probability of successfully altering at least 40% of Y' by a random derangement of scan flip-flops is zero as long as $N > \alpha / 0.4 = 375$, which can be easily satisfied by a design with a reasonable long scan chain.

TABLE I. EVALUATION OF THE WATERMARKING SOLUTION

Circuit	N	N_0	N_1	P_c	WT_{org}	WT_{wm}	$\Delta WT(\%)$
diffeq	305	141	164	7.98E-91	8039952	8336818	3.69
mle	323	165	158	1.42E-96	11134684	11390037	2.29
tseng	385	183	202	4.99E-115	10990496	11356970	3.33
B15	449	219	230	2.09E-134	110319565	111431425	1.01
B21	490	255	235	1.31E-146	181147289	184424095	1.81
valu	495	225	270	2.11E-147	43449702	44090045	1.47
pmac	590	293	297	7.62E-177	15345981	15638009	1.90
s15850	597	318	279	2.11E-178	81885476	82508775	0.76
s13207	669	328	341	1.50E-200	41259551	41947677	1.67
B22	735	378	357	2.54E-220	424594704	435128612	2.48
frisc	886	421	465	2.16E-265	168320151	167775831	-0.32
s38584	1426	715	711	<1.0E-300	693516003	703091069	1.38
s38417	1636	815	821	<1.0E-300	1770032478	1827785454	3.49
s35932	1728	849	879	<1.0E-300	102769343	105100815	2.27

V. CONCLUSION

This paper proposes a publicly detectable IP watermarking scheme. The ownership information is encrypted and transformed into constraints to reorder the scan chain for minimal switching power during test. The proposed method has overcome the vulnerability of previous power-driven scan chain watermarking scheme to enable the IP authorship to be verified publicly in the field without the risk of exposing the watermarked flip-flop locations. Experimental results have shown that the improved method has very low probability of coincidence and acceptably low test power consumption overhead.

REFERENCES

[1] A. B. Kahng *et al.*, "Constraint-based watermarking techniques for design IP protection," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Syst.*, vol. 20, no. 10, Oct. 2001, pp. 1236-1252..

[2] A. Cui, C. H. Chang and S. Tahar, "IP watermarking using incremental technology mapping at logic synthesis level," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Syst.*, vol. 27, no. 9, Sept. 2008, pp. 1565-1570.

[3] A. Cui and C. H. Chang, "Intellectual property authentication by watermarking scan chain in design-for-testability flow," in *Proc. IEEE Int. Symp. on Circuits and Syst.*, Seattle, USA, May 2008, pp. 2645-2648.

[4] D. Kirovski, and M. Potkonjak, "Intellectual property protection using watermarking partial scan chains for sequential logic test generation," in *Proc. IEEE High Level Design*, *Verification*, *and Test Conf.*, Nov. 1998.

[5] A. T. Abdel-Hamid, S. Tahar and E.M. Aboulhamid, "A survey on IP watermarking techniques," *Design Automation for Embedded Systems*, vol. 10, Springer Verlag, 2005, pp. 1-17.

[6] Y. Bonhomme, P. Girard, C. Landrault and S. Pravossoudovitch, "Power driven chaining of flip-flops in scan architectures," in *Proc. IEEE Int. Test Conf.*, Washington, USA, 2002, pp. 796-803.

[7] A. Menezes, P. van Oorschot and S. Vanstone, *Handbook of Applied Cryptography*. CRC Press, 1996.