A Watermarking System for IP Protection by a Post Layout Incremental Router

Tingyuan Nie
Department of Information Science
Kochi University
2-5-1 Akebono-cho, Kochi 780-8520 Japan
nieteien@is.kochi-u.ac.jp

Tomoo Kisaka
Department of Information Science
Kochi University
2-5-1 Akebono-cho, Kochi 780-8520 Japan
00ss019@is.kochi-u.ac.jp

Masahiko Toyonaga
Department of Information Science
Kochi University
2-5-1 Akebono-cho, Kochi 780-8520 Japan
toyonaga@is.kochi-u.ac.jp

ABSTRACT

In this paper, we introduce a new watermarking system for IP protection on post-layout design phase. Firstly the copyright is encrypted by DES (Data Encryption Standard) and then embedded by using an incremental router into the layout design. This watermarking technique uniquely identifies the circuit origin, yet is difficult to be detected or fabricated. The incremental router consists of a rip-up and a special re-router that inserts redundant bends into wires probabilistic. We evaluated the technique on various generated benchmark circuits to validate the completeness of the procedure. The results show it achieves almost 100% success for embedding with no extra area cost on design performances.

Categories and Subject Descriptors

B.7.2 [**Hardware**]: Integrated Circuits—*Design Aids*

General Terms

Design, Experimentation, Security

Keywords

Intellectual Property Protection (IPP), Post layout design, Incremental router, Watermarking.

1. INTRODUCTION

The progress of semiconductor manufacture process technology has made it possible to mount huge number of transistors on a chip such as 'system-on-chip' (SOC) that includes a perfect system on it. Thus a new design methodology like "design reuse" becomes more important. While the re-useable design, so-called IPP, is effective in reducing both the SOC design cost and TAT of development, the design method exposes the danger on security. As most IP designs need long time and many efforts for development or verification, the IP owners desire some guarantees that protect from the illegal use by the third person. Contrarily IP users also desire some guarantee that the contents they use are legal. The economic viability of this new IP based design paradigm is pending on the developments of techniques for IPP.

The research and implementation points of view show that IPP poses a unique set of new requirements that must be addressed mathematically yet practically. The watermarking technique is one of the solutions.

A traditional watermarking technique is viewed as an enabling technology to protect media data from unauthorized re-use. In general, watermarking enables ownership assertion, fingerprinting, authentication, integrity verification, content labeling, usage control and content protection. There also have been some feasible IPP techniques based on the watermarking [3-11]. The watermarking methods for IPP at the combinational logic synthesis and physical design were firstly proposed in 1998[3-4], and then emerge various concrete watermarking methodologies: pattern modification method [5], critical path preserving method [10], fingerprinting method [6], graph portioning method and so on. In recent years, various watermarking techniques for IPP have made a numerous progress.

However, previous watermarking methods for IPP require imposing much more than usual design restrictions, therefore the system becomes difficult to guarantee or optimize its performance of original IP design.

In this paper, we propose a watermarking technique and system at post-layout design stage which maintains the performance of original IP. The copyright information is firstly encrypted by DES (Data Encryption Standard) [2], and then the coded is embedded into circuits by a special incremental router [1] which keeps the size and the signal delays of original layout design. The code embedding may be failed due to high density of modern layout IP design. Therefore our system applies a procedure iteratively to achieve 100% success of watermarking. We evaluated our watermarking system by using several artificial benchmark circuits with various wire density and distributions, which resulted in a high success possibility of watermarking.

2. RELATED WORKS

Related work in watermarking and cryptography is represented in "IP Watermarking Techniques: Survey and Comparison" [11]. We therefore focus on related concepts within the physical design realm. There is no previous work of watermarking at post-layout design as far as we are concerned.

DAC 2005, June 13–17, 2005, Anaheim, California, USA.
Copyright 2005 ACM 1-59593-058-2/05/0006…$5.00.

However, constraint specification and management now receive close attention through all phases of chip implementation, including physical design. The most common constrains within the physical design include:

Timing constraints- Path delay constraints are often expressed in some form of Standard Delay Format (SDF), with heuristic "path cover" techniques used to reduce data volume and improve convergence of timing-driven layout tools. A static timing analysis engine may operate directly from the clock cycle times/offsets and I/O boundary timing to evaluate timing correctness, without explicit enumeration of timing path constraints. For the purposes of layout design, path delay constraints are typically budgeted into individual constraints on source-sink edges.

Physical (floor-planning) constraints- To improve timing convergence of the design process, assumptions made during RTL floor-planning or block floor-planning must be propagated to downstream flow stages (e.g., placement and global routing). This is often accomplished via region constraints: a given cell must be located in a given region of the layout. A set of cells must be co-located as a "group", etc. Such constraints may be captured using PDEF or equivalent formats which allow specification of assumed routing topology, layer usage, etc. at the level of global routing.

The mechanisms by which physical design tools enforce such constraints vary widely. However, classic paradigms generally do not support constraints well. The implications for watermarking in physical design are that: 1) current tools do not easily support too many "extra" watermarking constraints and 2) introduction of too many watermarking constraints will likely degrade solution quality. These issues complicate the choice of watermarking technique.

According to the recent VSI reports and general watermarking approaches, the watermarking router must satisfy the following issues.

1) There are no performance influences and area size changes to a layout design.

2) Impossible to distinguish whether external information is embedded or not by manually or mechanically only by the shape, almost similar to the original layout design.

3. POST-LAYOUT WATERMARKING

3.1 Overview

We illustrate outline of our watermarking system for IPP as shown in Figure1. The system is constructed of the following processes.

i) Encoding the copyright into bit string code by DES with a private key for owner and IP user.

ii) Assigning the encrypted bit codes to nets respectively by using cross-reference.

iii) Embedding the code into the IP layout design. A net corresponds to the code value '1' is applied our incremental router.

iv) Delivering IP to LSI layout designers.

v) Verifying the IP on LSI layout design by an entrusted silicon foundry or IP owner using the incremental router and DES decoder with the private key.

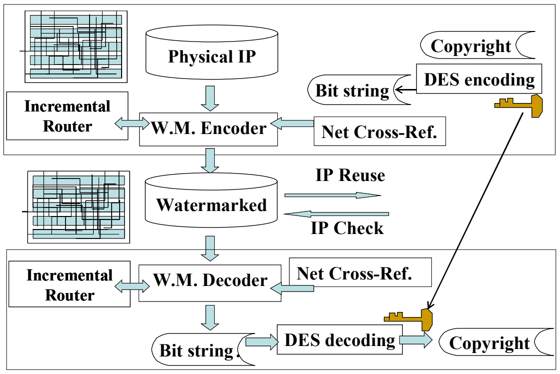
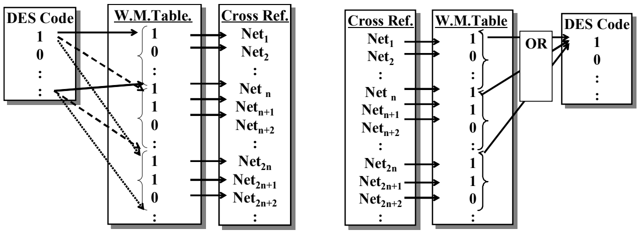


Figure1 Post-layout Watermarking System for IPP

3.2 Watermarking

We adopt the IP owner information and serial number as the signature source. The copyright code, bit unit string encrypted by DES, is assigned to more than one net by the cross-reference as shown in Figure2(a). The net corresponding to'1' in the bit code is applied our incremental router that insert redundant bend into the original wire shape. Here, this modification has no damage to the size and signal delays of design. However the success possibility of our incremental router depends on the density of wires of the layout. As a scheme, our system tries to assign the code to the nets as many as possible, i.e. the code of encrypted copyright is embedded into the layout design iteratively until to complete embedding. This iteration will make system robust.



(a) Assign DES code into nets (b) Extract code from net

Figure2 Cross-reference between encrypted code and nets

The encrypted code can also be extracted from the wires by using the same incremental router. The shape of wire is the same even if we applied the incremental router, then the net corresponds to '1', otherwise it corresponds to '0'. If we apply DES decoder to the extracted bit code with user's private key, then we can get the copyright signature. When the IP is illegally fabricated, the copyright signature will be broken and the illegal fabrication will be found. Those assurances can be used to defend the intellectual property. It provides an effective validation for IPP reuse.

3.3 Incremental Router

The embedding flow by the incremental router is shown in figure3. In case the incremental router could not find a path to insert an extra bends into the net assigned to '1', the router will keep the shape of the net and turn to watermark the next bit information and return a 'fail' message. If succeeded, success mark of this bit will be set to '1' and OR operated with pre-value. Once the bit information is successfully watermarked, it will be regarded as a success for this bit. If all bits of the coded information are successfully watermarked, the signature will be regarded as successfully be watermarked. We scale the system success probability with above rule. Iterative embeddings help the system enhance success possibility as described before.

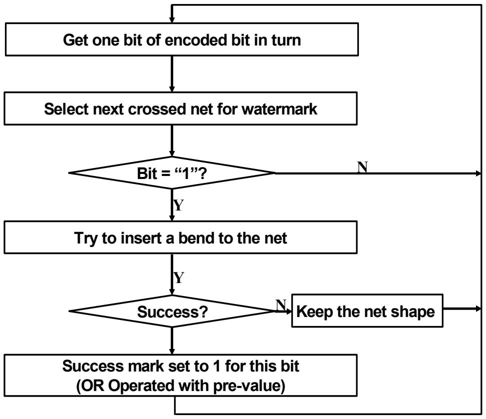

Get one bit of encoded bit in turn
Select next crossed net for watermark
Bit = "1"?
N
Y
Try to insert a bend to the net
Y
Success?
N
Keep the net shape
Success mark set to 1 for this bit
(OR Operated with pre-value)

Figure3 Embedding flow

Our incremental rip-up and re-router a net to insert an extra bend in the wire. Since it is no meaning for general router to include such a redundant bend, yet it can be used as a mark, while it seems difficult to distinguish to other wire patterns without this special router. Our incremental router is implemented by following steps based on the multilayer maze router.

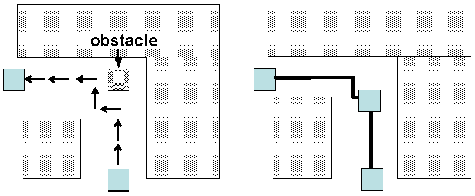


Figure4 Bend insertion illustration

Incremental Watermarking Router

Step1. Obtain an adequate corner points on the initial wire.

Step2. Rip the wires around the corner up.

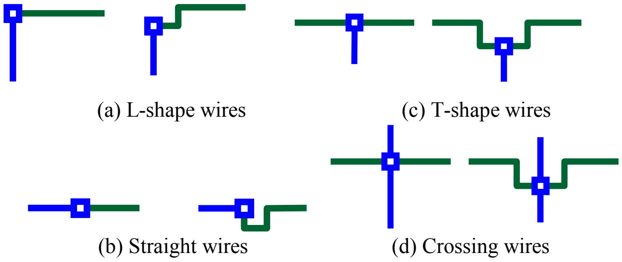
Step3. Set obstacles at the corner as shown in Figure4.

Step4. Try to route around the obstacles by multilayer maze router.

If it is successfully done, then go to Step7, otherwise then output 'fail' message.

Step7. Stop.

The selection of appropriate corner is based on the number of lines connected to via. The possible corner insertions patterns are shown in Figure5. We embedded a corner into the interconnection on the 'L' shape shown in Figure5 (a). Obviously the wire length is the same before insertion and after insertion. Generally, the detailed router seeks the shortest path between the terminal points, while our incremental router re-routers the terminals by setting barriers nearby the original interconnection point without through any via. This procedure applies to every interconnection in this net until it succeeded. Such operation makes the incremental router to re-route an extra bend near-by the original corner while does not change the wire length if the wire area is not very congested. Moreover, we will preserve the obstacles until the watermarking completeness in order to escape the interaction in the watermarking process, both in watermarking insertion and extraction.



(a) L-shape wires

(c) T-shape wires

(b) Straight wires

(d) Crossing wires

Figure5 Possible corner insertions

4. EXPERIMENTAL RESULTS

We evaluated our proposed watermarking system by conducting hundreds of experiments on machine with 2.4GHz Celeron CPU and 512MB main memory. Program is implemented in C language on Cygwin environment. The benchmark circuits are generated by routing the randomly distributed triplet-terminal nets and (or) pair-terminal nets with the multilayer maze router. Each benchmark insists of 576 terminals that support a few iterative embeddings. In our experience, the DES-coded bit list is '64 bit string', i.e. 8 characters. The circuits are fixed on the region of 150 × 150 grids.

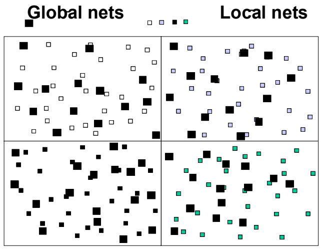


Figure6 Build benchmark with local and global nets

The terminal distribution of the benchmark process is shown in Figure6. We allocate two different ratios (R1, R2) for local nets and global nets; R1 is the ratio of terminals located in all of four divided local-net domains, while R2 is the ratio of terminals located in global-net domain. We setup the ratios of R1 and R2 (R1:R2) into three categories: (5:1), (3:1), and (1:1). As known, layout circuits are almost composed by triplet-terminal nets or pair-terminal nets. To evaluate our system adequately, we relatively add three benchmark types with different net distribution ratios, full triplet-terminal net circuit, full pair-terminal net circuit, and half-half circuit shown in table 1. In this way, we created 270 benchmarks for system evaluation. After performing several iterative embedding on all kinds of benchmarks, we obtained the system success probability results. We can see from table 1, the system average success probabilities exceeds 90.00%, and runtime is from 11.007 to 14.204 seconds. The system success probabilities for benchmark C21~C33 is higher than others due to more pair-terminal nets applied, which provides more chances to be watermarked than C11~C13. C31~C33's success probabilities approximate even up to 100%. It is optimistic for our post-design approach to watermark and protect an IP block.

Table 1 Experimental results for various benchmarks

Type	Local : Global	Pair:Triple	Avg. Prob.	Avg. time(Sec.)
C11	1 : 1	0:100	90.00%	13.071
C12	3 : 1	0:100	93.33%	11.671
C13	5 : 1	0:100	96.67%	11.007
C21	1 : 1	50:50	100.00%	14.204
C22	3 : 1	50:50	96.67%	13.612
C23	5 : 1	50:50	100.00%	13.342
C31	1 : 1	100:0	96.67%	12.544
C32	3 : 1	100:0	100.00%	11.596
C33	5 : 1	100:0	100.00%	11.250

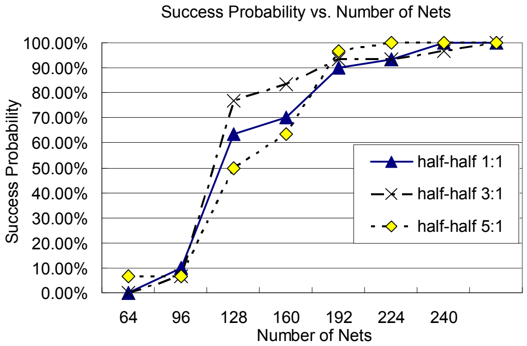


Figure7 Relationship between system probability and nets number

We illustrated how the success probability depends on the nets number about half-half type in Figure7. The success probabilities progress along with the net number increase. It indicates that the system success probability will achieve almost 100% if the embedding is supported by enough nets. Actually, to a certain signature, modern circuit can support many times' watermarking proportionally because of their millions of nets. In Figure8, we show two example layout patterns before and after watermarked.

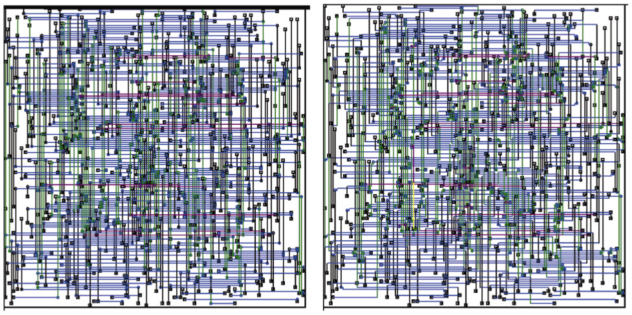


Figure 8 A post layout watermarking example

Our approach has certain resistance to several common attack schemes. For the embedding attacks, the intruder has to choose to have a ghost searching because he/she does not have our incremental router tool. It is at least as difficult as breaking a selected crypt protocol (DES), the probability Pc$\leq 2^{-56}$, which is considered to be a practically impossible task.

5. CONCLUSION

We proposed a post layout level watermarking system for IPP. The system consists of DES cryptography and an incremental router with iterative watermarking function. The experimental result shows that our system is reachable and reliable, and maintains the circuit's performance simultaneously. And we will carry out some experiences on practical blocks to identify our approach in recent work.

6. ACKNOWLEDGEMENTS

Thanks to Prof. Shigeo Kuninobu, Prof. Kennichi Shiota for their technical discussions. Authors would like to thank to RSP (Regional Science Promotion Program) of JST (Japan Science and Technology Agency) for their supports.

7. REFERENCES

[1] Tingyuan Nie, Masahiko Toyonaga, Ken-ichi Shiota, "A watermarking system for VLSI layout using a special router," 2003 Shikoku-section joint convention record of the institutes of electrical and related engineers (in Japanese), pp9-11 2003.

[2] DATA ENCRYPTION STANDARD (DES) http://www.itl.nist.gov/fipspubs/fip46-2.htm.

[3] D. Kirovski, Y.-Y. Hwang, M. Potkonjak and J. Cong "Intellectual Property Protection by Water-marking Combinational Logic Synthesis Solutions," Proc. ACM/IEEE International Conference on Computer Aided Design, San Jose, California, pp. 194-198, November 1998.

[4] A.B. Kahng, J.Lach, W. H. M-Smith, S.Mantik, I. L. Markov, M.Potkonjak, P. Tucker, H.Wang, G.Wolfe, "Constraint-Based Water-marking Techniques for Design IP Protection," IEEE Transactions on Computer-Aided Design of Integrated circuits and Systems, VOL. 20, NO. 10, pp. 1236-1252, October 2001.

[5] S.H. Kwok, C.C. Yang, K.Y. Tam, "Water-mark Design Pattern for Intellectual Property Protection in Electronic Commerce Applications", 33rd Hawaii International Conference on System Sciences-Volume 6, pp.6038, January 04 07, 2000.

[6] John Lach, William H. Mangione-Smith, Miodrag Potkonjak, "Fingerprinting Techniques for Field-Programmable Gate Array Intellectual Property Protection," IEEE Transactions on Computer-Aided Design of Integrated circuits and Systems, VOL. 20, NO. 10, pp. 1253-1261, October 2001.

[7] A. B. Kahng, J. Lach, W. H. Mangione-Smith, S. Mantik, I. L. Markov, M. Potkonjak, P. Tucker, H. Wang and G. Wolfe, "Water-marking Techniques for Intellectual Property Protection", 35th Design Automation Conference, pp.776-781, June 1998.

[8] Andrew B. Kahng, Stefanus Mantik, Igor L. Markov, Miodrag Potkonjak, Paul Tucker†, Huijuan Wang and Gregory Wolfe, "Robust IP Water-marking Methodologies for Physical design", 35th Conference on Design Automation Conference (DAC'98), pp. 782-787, June 15 19, 1998.

[9] Greg Wolfe, Jennifer L. Wong, and Miodrag Potkonjak, "Water-marking Graph Partitioning Solutions," 38th Conference on Design Automation (DAC'01), pp. 486-489, June 18 22, 2001.

[10] Miodrag Potkonjak, "Water-marking While Preserving The Critical Path", 37th Conference on Design Automation (DAC'00), pp. 108-111, June 05-09, 2000.

[11] Amr T. Abdel-Hamid, Sofi`ene Tahar, El Mostapha Aboulhamid, "IP Water-marking Techniques: Survey and Comparison", The 3rd IEEE International Workshop on System-on-Chip for Real-Time Applications (IWSOC'03), pp. 60, June 30 – July 02, 2003.