SOC TEST SCHEDULING ALGORITHM USING ENHANCED RECTANGLE PACKING

LEE SIAW CHEN

UNIVERSITI TEKNOLOGI MALAYSIA

SOC TEST SCHEDULING ALGORITHM USING ENHANCED RECTANGLE PACKING

LEE SIAW CHEN

A project report submitted in partial fulfillment of the
requirements for the award of the degree of
Master of Engineering (Electrical – Electronics and Telecommunication)

Faculty of Electrical Engineering
University Teknologi Malaysia

NOVEMBER 2009

ABSTRACT

The System-on-Chip (SoC) test scheduling algorithm based on rectangle packing was previously proposed by Iyengar *et al.* in 2002. This method had been proven its effectiveness on SoC test application time optimization. Xia *et al.* further improve the flexibility of rectangle packing approach by implementing the distributed rectangle bin-packing approach which allows core wrapper pins from one particular core to be assigned to non-consecutive SoC Test Access mechanism (TAM) through vertical partitioning of core rectangles. However, the above mentioned methods still result in significant idling time. Therefore, this project proposes a new scheduling method, namely the enhanced rectangle packing, which is an extension to the original rectangle packing and distributed rectangle packing. The proposed algorithm horizontally partitioned the core rectangles to obtain rectangles of smaller size for idling time reduction, which in turn successfully shorten the total test application time for a SoC. Experimental results conducted on ITC'02 SoC test benchmark circuits show the effectiveness of the enhanced rectangle packing algorithm in reducing SoC test application time for up to 6% in maximum.

ABSTRAK

Algoritma penjadualan untuk pengujian sistem-dalam-cip (SoC) yang berasalkan bungkusan-persegi telah diperkenalkan oleh Iyengar dan kumpulan kajiannya pada tahun 2002. Algoritma ini merupakan kaedah yang amat berkesan dalam usaha untuk mengurangkan jumlah masa yang diperlukan dalam pengujian SoC. Xia dan kumpulan kajiannya pula mengemukakan algoritma bungkusan-persegi-teredar yang melonggarkan sekatan algoritma bungkusan-persegi. Bungkusan-persegi-teredar membenarkan peruntukan mekanisma-akses-ujian (TAM) tak-berterutan bagi sesuatu modul dengan memperkenalkan potongan-melintang untuk persegi-persegi yang mewakili modul-modul di dalam sesebuah SoC. Akan tetapi, kedua-dua cara yang diperkenalkan diatas masih meninggalkan ruangan kosong di dalam jadual pengujian SoC. Ruangan kosong ini merupakan masa yang dibazirkan semasa pengujian SoC, di mana tiada mana-mana modul yang sedang menggunakan TAM tersebut pada masa itu. Oleh yang demikian, projek ini mengemukakan satu pendekatan baru yang boleh mengurangkan ruang-ruangan kosong ini dan seterusnya memendekkan jumlah masa yang diperlukan bagi pengujian sesuatu SoC. Kaedah baru ini dinamakan bungkusan-persegi-tambahan. Ia merupakan pengingkatan daripada bungkusan-persegi and bungkusan-persegi-teredar. Bungkusan-persegi-tambahan membolehkan potongan-menegak atau potongan-melintang atau kedua-duanya sekali bagi persegi-persegi untuk menghasilkan persegi-persegi yang bersize kecil. Persegi-persegi kecil ini boleh dimasukkan dalam ruangan kosong bagi jadual pengujian sesebuah SoC. Dengan itu, ruangan kosong di dalam jadual pengujian dapat dikurangkan dan seterusnya memendekkan jumlah masa pengujian SoC tersebut. Menurut kepada keputusan kajian yang dijalankan atas sampel litar ITC'02, keadah bungkusan-persegi-tambahan yang dikemukakan oleh project ini telah berjaya untuk mengurangkan jumlah masa pengujian sebanyak 6% secara maksimumnya.

TABLE OF CONTENTS

CHAPTER	TITLE		PAGE
	DECLARATION		ii
	DEDICATION		iii
	ACKNOWLEDGEMENTS		iv
	ABSTRACT		v
	ABSTRAK		vi
	TABLE OF CONTENTS		vii
	LIST OF TABLES		xi
	LIST OF FIGURES		xii
	LIST OF ABBREVIATIONS		xiv
	LIST OF SYMBOLS		xv
	LIST OF APPENDICES		xvi
1	**INTRODUCTION**		**1**
	1.1	Background and Motivation	1
	1.2	Previous Work	2
	1.3	Problem Formulation	4
	1.4	Objectives	5
	1.5	Scope of work	6
	1.6	Research Contribution	7

	1.7	Thesis Outline	7
	1.8	Summary	8
2	**SYSTEM-ON-CHIP (SOC) TESTING OVERVIEW**		**9**
	2.1	System-on-Chip (SoC) Testing Background	9
	2.2	Core Wrapper	11
		2.2.1 TestShell	13
		2.2.2 IEEE 1500 Standard for Embedded Core Test (SECT)	16
		2.2.3 TestCollar	18
		2.2.4 Combinational Approach	18
	2.3	Test Access Mechanism (TAM)	19
		2.3.1 Multiplexed Architecture	20
		2.3.2 Distributed Architecture	20
		2.3.3 Daisy-chained Architecture	21
		2.3.4 Fixed-width TestBus Architecture	22
		2.3.5 Flexible-width Architecture	23
	2.4	Test Scheduling	24
	2.5	Summary	25
3	**RECTANGLE PACKING FOR SYSTEM-ON-CHIP (SOC) TEST SCHEDULING**		**26**
	3.1	Core Test Time Optimization with Fixed TAM Width	26
	3.2	Core Test Time Optimization with Flexible TAM Width	27
		3.2.1 *Design_wrapper* Algorithm	28
		3.2.2 Pareto-optimal Width	31
	3.3	The Rectangle Packing Modeling for	

System-on-Chip (SoC) Test Scheduling 32
3.3.1 Distributed Rectangle Packing 34
3.3.2 Preferred TAM Width Calculation 35
3.4 ITC'02 SoC Test Benchmarks Circuits 36
3.5 Summary 39

4 PROPOSED ENHANCED RECTANGLE PACKING ALGORITHM 40
4.1 The Basic Concept of the Enhanced Rectangle Packing 40
4.2 Horizontal Partitioning of Core Rectangle 43
4.3 Wrapper Modification 44
4.4 The Enhanced Rectangle Packing Algorithm 46
4.4.1 Wrapper Design Optimization 48
4.4.2 Horizontal Partitioning and Preferred Width Calculation 49
4.4.3 Schedule Core(*ia*) with Preferred TAM Width 49
4.4.4 Heuristic 1: Schedule Core(*jb*) with Preferred TAM Width 50
4.4.5 Heuristic 2: Rectangle Insertion in Idle Time 51
4.4.6 Heuristic3: Increasing TAM Width to Fill-up the Idling Time 51
4.4.7 Declaration of Idle Test Slot and the Completion of Core Test 51
4.4.8 Horizontal Partitioning Check 52
4.5 Summary 52

5	**EXPERIMENTAL RESULT AND DISCUSSION**		**54**
	5.1	Experimental Result and Discussion	54
	5.2	Summary	59
6	**FUTURE WORK AND CONCLUSION**		**60**
	6.1	Recommendation for Future Work	60
	6.2	Conclusion	62
REFERENCES			**63**
APPENDIX A			**66**

CHAPTER 1

INTRODUCTION

This thesis proposes an algorithm to determine the core tests ordering of a system-on-chip (SoC) which permits shorter SoC test application time. The suggested enhanced rectangle packing test scheduling algorithm is implemented using Microsoft C++ object oriented programming language. The previous method, rectangle packing algorithm is also developed in the same language. Both algorithms are applied to eleven ITC'02 SoC test benchmarks for test time comparison purpose. This chapter introduces the research background, related previous work, motivation and problem statements, objectives, scope of work and the research contribution. The organization of this thesis is summarized at the end of the chapter.

1.1 Background and Motivation

SoC or a core-based system is comprised of a set of cores (also referred as modules, blocks, or macros). Cores are simply some well-partitioned, pre-defined and

pre-verified pieces of logic to be combined with glue logic to become the designed complex system. Cores may, for instance, be digital signal processing (DSP) block, central processing unit (CPU), memory array, high speed signal interface (HSSI) module, and any user-defined logic macros. They can be reused design from previous projects or bought over from core vendors [1].

SoC has emerged to be the most popular integrated circuit (IC) design approach nowadays. The rapid growth of SoC design complexity has made the development of SoC test solution thorny, at the same time led to significant raise of SoC test application time. Consequently, semiconductor industry is urging for an efficient SoC test solution to ease the test development process besides cutting down the overall SoC test application time for maximum test cost reduction.

1.2 Previous Work

In general, SoC test related issues include the optimal design of test wrapper and test access mechanism (TAM), and SoC test scheduling algorithm for test application time optimization. Many research works have been carried out in recent years to address these issues.

Most of the earlier papers proposed methods to solve the wrapper design and the TAM design independently. It's not until 2002 when Iyengar et al. [2] first introduced the approach to cater test wrapper and TAM design optimization in-conjunction by proposing an integrated wrapper/TAM design co-optimization which has effectively balanced out the wrapper scan chain length for test application time optimization in any

given TAM width. Zou et al. [3] then further improved the test wrapper design by considering core without internal scan chains.

With the establishment of balanced wrapper scan chain concept for test wrapper and TAM design, more focus has been put on the optimization of SoC test scheduling algorithm and the co-optimization of wrapper design and test scheduling. In [2], the co-optimization technique is based on the combination of integrated linear programming (ILP) with exhaustive enumeration. However, this method was limited to fixed TAM width configuration which is rigid and frequently leads to expensive CPU processing time. Thus, they formulated the SoC test scheduling as a 2-dimensional bin-packing problem (rectangle packing problem) for more flexible TAM width configuration which consume less CPU time [4]. However, the SoC test application time for most of the benchmark circuits was higher than [2]. Xia et al. [5] introduced the algorithm that realized vertical rectangle splitting. Smaller rectangles produced by this splitting are able to fill up the idling time resulted from [4] and thus reducing the SoC test application time. However, due to its restriction of only vertical splitting being allowed, the rectangles might result in many idle test slots.

Rectangle packing algorithm with TAM width flexibility has inspired the following research works that produce more optimized test scheduling algorithm. Random insertion (RAIN) scheduling algorithm introduced by Im et al in [6] achieved its optimization by random position insertion onto the sequence pair representation of rectangle placement. Ahn and Kang engaged ant colony optimization (ACO) approach to tackle the test scheduling problem that was formulated as a rectangle bin-packing problem [7]. But, little improvement was obtained compared to [6]. Yu et al. [8] proposed a two-stage genetic algorithm (GA) to solve the test scheduling problem that has been formulated into a two-dimension floor-plan problem with sequence pair architecture. It shows shorter test application time for two of the ITC'02 SoC benchmark circuits compared to [4] and [3]. A GA model for optimal test scheduling was then

proposed in [9] where the differential evolution and self-adaptive mutation were brought into the traditional genetic algorithm. However, no experiment is conducted on ITC'02 benchmark.

Besides TAM width constraint, Huang et al. [10] extended the rectangle representation concept into a three-dimensional bin-packing problem to include power constraints on the third axis and proposed a best-fit heuristic method to solve the problem. In [5], Xia et al. presented the algorithm of distributed rectangle packing for non-consecutive SoC TAM allocation to a core in the same time addressed the test scheduling issue under power constraints using sequence pair representation together with evolutionary algorithm. While Wu et al. [11] applied the B*-tree based floor-planning technique to address the similar issue, Harmanani and Farah in [12] proposed an optimization method based on simulated annealing which can handle precedence constraints that preserve the desirable test ordering for SoC test scheduling with or without power constraints. Both [14] and [15] are taking design hierarchy constraints into consideration for SoC test scheduling optimization where [14] is based on simulated annealing technique and [15] is engaging GA approach.

1.3 Problem Formulation

This research is motivated by an improved memory allocation method introduced in [20] where the memory allocation problem is modeled by bin-packing algorithm with split-able items. Hence, in this thesis, the test scheduling problem is modeled as rectangle bin-packing problem with split-able rectangle. This work is an extension from rectangle packing in [2]. Different from [2] and [4], the rectangles in this research are split-able horizontally and vertically.

The SoC test scheduling problem that we address in this thesis is as follows.

Definition 1 (SoC test scheduling problem)

Input: A SoC benchmark with its test parameters including:

1) The number of cores in the SoC,
2) The number of primary inputs, primary outputs, bidirectional I/Os, test patterns, internal scans chains, and scan chain lengths for each core.

Determine: 1) The optimum TAM width for each core,
2) The ordering of core test, and
3) The assignment of core TAM to SoC TAM partition.

Output: 1) The optimum SoC TAM width, and
2) The SoC test schedule with minimized test application time associated with the optimum SoC TAM width.

1.4 Objectives

The overall objective of this research is to develop the Enhanced Rectangle Packing test scheduling algorithm for shorter SoC test application time. This objective is modularized into the following sub-objectives.

1) To design the Enhanced Rectangle Packing test scheduling algorithm based on split-able bin-packing concept.
2) To compare the total SoC test application time from enhanced rectangle packing algorithm over that of rectangle packing algorithm.

1.5 Scope of Work

1) The enhanced rectangle packing algorithm is designed and implemented in Microsoft Visual C++ Object Oriented programming language.
2) The rectangle packing algorithm is implemented in Microsoft Visual C++ Object Oriented programming language.
3) The effectiveness of algorithm is verified and compared over previous method on ITC'02 SoC test benchmarks.
4) The proposed algorithm considers only testing time optimization without taking care of the power limitation.
5) This algorithm involved only structural core test with no consideration on functional core test as well as the testing for interconnect.
6) The algorithm assumes flatten SoC hierarchy is engaged during test mode whereby all cores are located in the same level.
7) There is no dependency between cores during testing whereby each core can be tested independently and concurrently.
8) The flexible test bus TAM architecture is employed in this project.

1.6 Research Contribution

This thesis proposes a new algorithm to resolve SoC test scheduling problem. The enhanced rectangle packing algorithm is an extension from rectangle packing [4]. The suggested algorithm introduces flexible core partitioning which allow horizontal and vertical splitting of core rectangle representation. As a result of the capability of flexible core splitting, SoC test application time is successfully cut down by three to four percent for most of the benchmarks compared to the previous solution.

In addition to this, since all the optimization algorithms suggested in previous works (as mentioned in section 1.2) are developed on top of the rectangle packing algorithm proposed in [4], we believe that those works can achieve better result if they are redone based on the enhanced rectangle packing algorithm suggested in this thesis.

1.7 Thesis Outline

The work in this thesis is conveniently organized into six chapters. Chapter 1 first presents the background and previous related works follows by motivation and problem formulation, research objectives, scope and contribution, before concludes with thesis organization. Chapter 2 briefly reviews on the basic theory related to SoC background which includes the core wrapper architecture and the test scheduling mechanism. The third chapter introduces the rectangle packing approach employed in SoC test scheduling, which served as the fundamental concept of the enhanced rectangle packing proposed in this thesis. Chapter 4 gives the details on the proposed enhanced rectangle packing algorithm while the experimental results are discussed in Chapter 5. In

the final chapter of this thesis, the research work is summarized and deliverables of research are presented. Suggestion for potential future extensions and improvement to SoC test scheduling is also stated.

1.8 Summary

In this chapter, an introduction was given on the background and motivation of the research. The need for an efficient test scheduling for SoC test solution was discussed and several research objectives and scope had been identified. The following chapter will provide brief summaries on relevant background theories and literature reviews.

REFERENCES

1. Erik Larsson. *Introduction to Advanced System-on-Chip Test Design and Optimization*. Dordrecht, The Netherlands: Springer. 2005.
2. V. Iyengar, K. Chakrabarty and E. J. Marinissen. Test Wrapper and Test Access Mechanism Co-optimization for System-on-Chip. *Proc. JETTA*. 2002. 18: 213–230.
3. Wei Zou, Sudhakar M. Reddy, Irith Pomeranz, Yu Huang. SOC Test Scheduling using Simulated Annealing. *Proc. VTS*. 2003. 253-258.
4. V. Iyengar, K. Chakrabarty and E. J. Marinissen. On using Rectangle Racking for SoC Wrapper/TAM Co-optimization. *Proc. VTS*. 2002. 253-258.
5. Yu Xia, Malgorzata Chrzanowska-Jeske, Benyi Wang and Marcin Jeske. Using a Distributed Rectangle Bin-packing Approach for Core-based SoC Test Scheduling with Power Constraints. *ICCAD*. 2003. 100-105.
6. Jung_Been Im, Sunghoon Chun, Geunbae kim, Jin-Ho An, Sungho Kang. RAIN (Random Insertion) Scheduling Algorithm for SoC Test. *Proceedings of the 2004 Asia Test Symposium*. 2004.
7. Jin-Ho Ahn, Sungko Kang. SoC Test Scheduling Algorithm using ACO-based Rectangle Packing. *Proc. 2006 ICIC*. 2006. 655-660.
8. Y. Yu, X. Y. Peng, Y. Peng. A Test Scheduling Algorithm based on Two-stage GA. *Proceedings of the 2006 J. Physics.* 2006. 48: 658-662.
9. Zhan Jinyu, Li Xunsheng, Guo Bing, Xiong Guangze and Sang Nan. A Test Scheduling Scheme for Core-Based SoCs Using Genetic Algorithm. *Proceedings of the 2008 ICESS*. 2008.

10. Y. Huang, S. M. Reddy, W.-T. Cheng, P. Reuter, N. Mukherjee, C.-C. Tsai, O. Samman, Y. Zaidan. Optimal Core Wrapper Width Selection and SOC Test Scheduling based on 3-D Bin Packing Algorithm. *Proceedings of the 2002 International Test Conference.* 2002. 74-82.
11. Jen-Yi Wuu, Tung-Chieh Chen, and Yao-Wen Chang. SoC Test Scheduling Using the B*-Tree Based Floorplanning Technique. *Proc. ASP-DAC*. 2005. 2: 1188-1191.
12. Haidar M. Harmanani and Rana Farah. Integrating Wrapper Design, TAM Assignment, and Test Scheduling for SOC Test Optimization. *Proceedings of the 2002 International Test Conference.* 2008. 149-152.
13. I. Ghosh, and M. Fujita. Optimizing SOC Test Resources using Dual Sequences. *Proc. International Federation for Information Processing*. 2006. 200: 181-196.
14. Haidar M. Harmanani and Rana Farah. Integrated Test Scheduling, Wrapper Design, and TAM Assignment for Hierarchical SOC. *Proceedings of MWSCAS.* 2007. 1388-1391.
15. Chandan Giri, Dilip Kumar Reddy Tipparthi and Santanu Chattopadhyay. Genetic Algorithm based Approach for Hierarchical SOC Test Scheduling. *Proceedings of ICCTA.* 2007. 141-145.
16. Y. Zorian, E. J. Marinissen, and S. Dey. Testing Embedded-Core Based System Chip. *Proceedings of International Test Conference (ITC).* Washington, DC, USA. October 1998. 130-143.
17. E.J. Marinissen, R. Arendsen, G. Bos, H. Dingemanse, M. Lousbera, and C. Wouters. A Structured and Scalable Mechanism for Test Access to Embedded Reusable Cores. *Proceedings of International Test Conference (ITC).* Washington, DC, USA. October 1998. 284-293.
18. IEEE Standard 1149.1. Joint Test Action Group. 2001.
19. E. J.Marinissen, Y. Zorian, R. Kapur, T. Taylor, and L. Whetsel. Towards a Standard for Embedded Core Test: An Example. P*roceedings of International Test Conference (ITC)*. Atlantic City, NJ, USA. September 1999. 616-627.

20. Francisco da Silva, Teresa McLaurin and Tom Waayers. *The Core Test Wrapper Handboook – Rationale and Application of IEEE Std. 1500*. United State of America: Springer. 2006.
21. P. Varma and S. Bhatia. A Structured Test Re-Use Methodology for Core-Based System Chips. P*roceedings of International Test Conference (ITC)*. Washington, DC, USA. October 1998. 294-302.
22. J. Aerts and E. J. Marinissen. Scan chain Design for Test Time Reduction in Core-Based ICs. P*roceedings of International Test Conference (ITC)*. Washington, DC, USA. October 1998. 448-457.
23. K. Chakrabarty. Optimal Test Access Architectures for System-on-a-Chip. *Proc. ACMTrans. Design Automation of Electronic Systems*. Jan 2001. 6: 26–49.
24. V. Iyengar and K. Chakrabarty. Test Bus Sizing for System-on-a-Chip. *Proc. IEEE Trans. Computers.* May 2002. 51.
25. E. J. Marinissen, S.K. Goel and M. Lousberg. Wrapper Design for Embedded Core Test. P*roceedings of International Test Conference (ITC)*. 2000. 911-920.
26. E. G. Coffman, Jr., M. R. Garey, D. S. Johnson and R. E. Tarjan. Performance Bounds for Level-oriented Two-dimensional Packing Algorithm. *Proc. SIAMJ. Computing.* 1980. 9: 809-826.
27. ITC'02 SoC Test Benchmark Circuit. *http://itc02socbenchm.pratt.duke.edu*. 2002.
28. Erik Jan Marinissen, Rohit Kapur, Maurice Lousberg, Teresa McLaurin, Mike Ricchetti and Yervant Zorian. On IEEE P1500's Standard for Embedded Core Test. *Proc. JETTA.* 2002. 18: 365-383.
29. Erik Larsson and Zebo Peng. An Integrated Framework for the Design and Optimization of SoC Test Solutions. *Proc. JETTA.* 2002. 18: 385-400.
30. Haidar M. Harmanani and Hassan A. Salamy. On Power-Constrained System-On-Chip Test Scheduling Using Precedence. P*roceedings of IEEE North-East Workshop*. 2006. 125-128.
31. Leah Epstein and Rob van Stee. *Improved Results for a Memory Allocation Problem*. Lecture Notes in Computer Science. Springer. 2005. 362-373.