Genetic Algorithm based Approach for SOC Test Scheduling

Mary Catherine
Assistant Professor
SCMS School of Engineering & Technology, Karukutt

Jagadeesh Kumar P
Assistant Professor
Govt. Model Engineering College
Thrikkakara, Kochi

ABSTRACT
In a Core based SoC design various Intellectual Property (IP) cores are integrated on a single chip called System on chip(SoC). The testing of this SoC is complex, resulting in a long test application time. But the testing time cannot be too long as the relevant cost will increase rapidly. The testing time can be minimized if an effective scheduling of the tests is done. This paper presents a test scheduling scheme for a core based SoC. The SoC test scheduling is an NP complete problem and hence this paper presents Genetic algorithm based test scheduling scheme for a core based SoC so as to minimize the testing time incorporating the power dissipation constraints. Genetic algorithm generate solutions to the scheduling problem using techniques inspired by natural evolution. The optimal solution obtained by running genetic algorithm is applied to the ITC'02 SoC test Benchmarks and provides minimum test time results.

General Terms
Intellectual Property(IP), System on chip(SoC), Scheduling Algorithm
Keywords
Core based SoC; test scheduling; test application time; SoC test Benchmarks.

1. INTRODUCTION
component in the cost of designing, manufacturing and maintaining in the present semiconductor industry.So testing such a complex SoC is a major issue in SoC Design in terms of test application time. In order to minimize the testing time, efficient test scheduling is important. Test Scheduling is a process that determines the test start and finishing time of every core in SoC such that overall test application time is minimized[2].The SoC Test Scheduling which is an NP complete problem can be optimized with Approximation algorithms and Heuristics. Both approximation algorithms and heuristic techniques do not always guarantee an optimal solution to an NP complete problem. This problem can be dealt with the traditional optimization approaches like Integer Linear Programming [5], Simulated annealing [6],Ant Colony Optimization [7] etc.This paper proposes a simple Genetic Algorithm to obtain minimum test application time under power dissipation constraints for a core based SoC with flattened design hierarchy. Compared to the approach for similar problem discussed in[1],this paper incorporates the power dissipation constraints while scheduling. Also here optimal solution obtained by running GA are validated using the new set of ITC'02 SoC test benchmarks, whereas in [1] it is validated with older version(ISCAS'85 and ISCAS '89) benchmarks.In this paper it is assumed a flattened design hierarchy for a core based SoC which is a simple approach in which top level and sub cores are at the same level which in turn simplifies the calculation of test time compared to the approach discussed in [2].The work presented here is the formulation of the SoC test scheduling problem based on genetic algorithm, optimization of the problem by running the algorithm and applying the optimal solution to the ITC'02 SoC test benchmarks so as to compute the test application time. It is found that better test time results are obtained for different TAM(test access mechanism)widths using this method. The paper also presents minimum test time results satisfying the power constraint criteria for SoC h953 which is the only benchmark among the ITC'02 benchmark set which has power dissipation numbers included. The rest of the paper is organized as follows. The recent works in SoC Test Scheduling using Genetic Algorithm is described in the next section. The Problem Formulation for Soc Test Scheduling is presented in section III. The design steps and system block diagram is dealt in section IV. The experimental results are given in section V. Finally the paper is concluded in Section VI.

2. RELATED WORK
The work by Zhan Jinyu et.al. [1] develops a Genetic algorithm model for optimal test scheduling with the differential evolution and self-adaptive mutation concepts incorporated,which can provide more stable and optimal solutions. The differential evolution strategy is utilized to obtain the next generation.Chandan Giri et.al. [2] proposed the testing of Hierarchical SoC's with multiple levels of embedded cores using genetic algorithm approach. The objective of the work was to minimize the testing time for a Hierarchical SoC.The problem was formulated using GA,algorithm was developed and testing time for various SoC's were compared. Chandan Giri,Soumojit Sarkar and Santanu Chattopadhyay proposed GA to co-optimize test scheduling and wrapper design under power constraints for core based SoC's [3]. SoC Test scheduling algorithm based on two dimensional Rectangle Bin Packing algorithm is also utilized.

3. PROBLEM FORMULATION
The SoC Test scheduling can be divided into two problems[1],with the power dissipation constraints:

(1) Optimal assignment of cores to test buses

(2) Optimal distribution of TAM(test access mechanism) width.

Problem 1: Minimizing the system testing time for a given SoC with Nc cores and NB test buses with width w1 , w2 ,..., WNB respectively, by optimally assigning the cores to the test buses.

Problem 2: Minimize the system testing time for a given SoC with Nc cores and NB test buses with total width W, by optimally allocating the total width among the NB buses and assigning cores to test buses.The constraint observed in SoC Test scheduling is:

• Power dissipation constraint

Power constraint criteria is, we calculate the sum of the maximum peak power for all the cores at a given time instance. It is assumed that for the entire test time of the core, maximum peak power is the same. The combined power dissipation must not exceed maximum power limit called the SoC Power Budget.

4. GA DESIGN AND SYSTEM BLOCK DIAGRAM

In GA, from the initial population parents are selected to perform the recombination and mutation operations to produce off springs. The offsprings undergo the feasibility check and surviving offsprings are then inserted to the population forming the new population. This evolution process continues for a few generations and terminates when optimal solution is reached.
The SoC Test scheduling problem discussed in the previous section is formulated with GA in 5 design steps.

(1) Encoding the chromosome

(2) Calculation of Fitness Measure

(3) Selection scheme for parent chromosomes

(4) Genetic Operations

(5) Feasibility Check and Convergence

4.1 Encoding the Chromosome

The binary encoding is used in the design for each of the chromosomes. The variables of the parameter set are concatenated in turn, namely N_B test buses of the SoC from No.1 gene onwards followed by the assignment of N_C cores of the SoC to the buses. For example, the chromosome in Figure 1 represent the bus width of the No.1, No.2 and No. N_B as 1110,0010 and 0101 respectively and the No.1 core, the No.2 core, the No.N_C core are respectively assigned to the No.2 bus,No.1 bus and No.1 bus.

4.2 Fitness Function

The Fitness of the chromosome is measured in terms of the cost of a solution, which is the total test time required to test

1110	0010	.....	0101	0010	0001	.....	0001
1	2		NB	NB+1			NB+NC

Fig 1:Chromosome structure

all cores in the system and is explained as follows[1]. Definition 1: If SoC design consist of N_C cores, and let core i has n_i inputs including data inputs and scan inputs and m_i outputs including data outputs and scan outputs, then core test width is defined as follows:

$$\Phi = \max(ni, mi) \quad (1)$$

Definition 2: If IP core i contains fi flip flops and Ni internal scan chains, then the required test cycles ti for testing core i is defined as follows: (where pi is the number of test patterns)

$$ti = pi \text{ (for combinational cores)}$$

$$ti = (pi + 1)\lceil fi / Ni \rfloor + pi \text{ (for sequential cores)} \quad (2)$$

Definition 3: If core i is assigned to bus j, the testing time Tij for core i can be computed as follows:

$$Tij = ti, \Phi \leq wi \quad (3)$$

$$Tij = (\Phi - wi + 1)\, ti, \Phi \geq wi \quad (4)$$

Let xij be a 0-1 variable defined as follows: if the core is assigned to bus j xij = 1

Otherwise xij = 0 (5)

For each bus, add all the testing times of the cores assigned to it. So the testing time of all cores assigned to test bus j will be

$$\sum_0^{Nc} Tij\ xij \quad \text{test cycles} \quad (6)$$

The system testing time is the maximum testing time of all buses, ∑Tij xij, since all test buses assumed to be used simultaneously for testing. Therefore the fitness function for the genetic algorithm will be defined as follows:

$$\text{fitness}(x) = \max j\{\sum_0^{Nc} Tij\ xij\} \quad (7)$$

4.3 Selection Scheme for Parent Chromosomes

The fitness measure of the chromosomes are calculated and the probability of each of them to be selected as parents is based on fitness proportionate values.Roulette wheel selection is used for selecting the parent chromosomes.Figure2 shows the roulette wheel representation of fitness values of all encoded chromosomes.

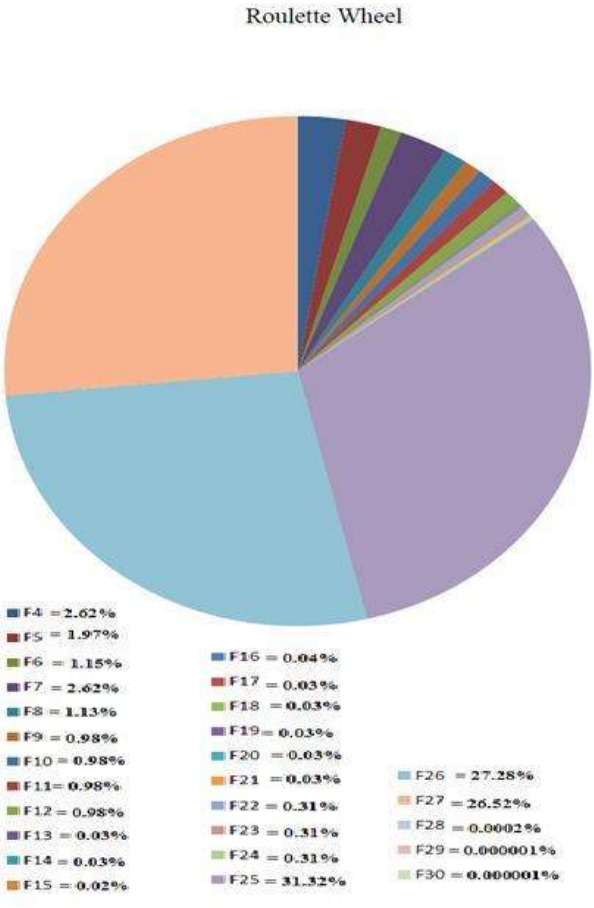

Fig 2: Example Roulette wheel representation of fitness

4.4 Genetic Operations

Crossover and Mutation are the two commonly used genetic operators. Crossover selects gene from parent chromosomes and create new offsprings. In this design, Single Point crossover is performed using a crossover mask of length same as that of chromosome length. Single Point Mutation operation is performed to introduce genetic diversity by simply flipping the bits at that position

4.5 Feasibility check and Convergence of GA

The Feasibility Check is performed for each gene of the chromosome satisfying the two criteria as listed below:

(1) The sum of the values of gene 1,gene 2,...gene NB should not exceed the TAM width under consideration.

(2) Checking for each field from gene NB to gene NB+NC for the assignment of cores to valid buses only.

The Convergence of GA happens when the best possible solution is obtained after iterating through 100's of generations. In this design it is based on when the algorithm continues with the same best chromosome for X number of iterations.

4.6 System Block Diagram

The figure 3 shows the system block diagram where genetic algorithm is used to provide the optimal solution for test scheduling problem. This optimal solution which describes the assignment of cores to the buses and optimal partition of TAM width can be applied to the ITC'02 Benchmark circuits[4]. The test application time can then be calculated, which will result in minimum test time under power constraints.

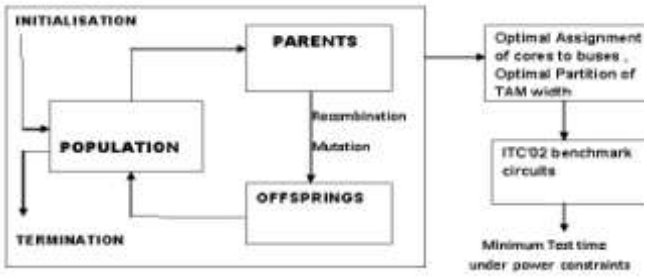

Fig 3: System Block Diagram

5. EXPERIMENTAL RESULTS

The proposed Genetic Algorithm for SoC Test Scheduling problem is implemented in C and is compiled and simulated using GCC compiler. The results has been validated using ITC'02 SoC Test Benchmarks. All tests were conducted on a 2.69 GHz AMD Athlon 64x2 Dual Core Processor with 1.75 GB memory. Here the results for 4 benchmark SoC's h953, a586710, d695 and g1023 are presented.

Table 1: Results for soc h953(8 cores) with different bus widths under power constraints

TAM width	Optimal partition	Optimal bus assignment	Test time(cycle)	Power
16	(15,1)	(2,2,2,1,1,2,1,	218802	$11.3x10^9$
24	(18,6)	(2,2,2,1,1,1,2, 2)	181263	$9.4x10^9$

Table 2: Best assignment for SoC h953 with different random seeds

Random seed	TAM width	Optimal partition	Optimal assignment
1	16	(3,13)	(1,1,1,2,2,2,1,1)
2	16	(5,11)	(1,1,1,1,2,2,2,2)
3	16	(9,7)	(1,1,1,1,1,1,2,1)
4	16	(4,12)	(2,1,1,2,2,2,2,2)
5	16	(13,3)	(1,2,1,2,2,1,2,2)
6	16	(4,12)	(2,2,2,2,1,1,1,1)

Table 3: Results for soc a586710 (7 cores) with different bus width

TAM width	Optimal partition	Optimal bus assignment	Test time(cycle)	Processor time
16	(3,13)	(1,1,1,1,2,1,1)	15285881	0.01 s
24	(9,15)	(1,2,1,1,2,2,1)	14474301	0.03 s

.

Table 4: Results for soc d695(10 cores) with different bus widths

TAM width	Optimal partition	Optimal bus assignment	Test time(cycle)	Processor time
16	(5,11)	(2,2,1,2,1,1,2,1,1,1)	28232	0.01s
24	(20,4)	(2,2,2,2,1,1,1,1,1,2)	24154	0.04s

Table 5: Results for soc g1023(14 cores) with different bus widths

TAM width	Optimal partition	Optimal bus assignment	Test time(cycle)	Processor
16	(9,7)	(2,1,2,2,2,1,1,1,2,1,2,1,	27145	0.04
24	(3,21)	(2,1,1,2,1,1,2,1,1,1,1,1,	22780	0.07

The optimal partition of test width and test bus assignment for different TAM widths for SoC h953 is shown in Table 1. The test time and power dissipation values (which satisfies

Power constraint criteria) is also shown in Table 1.Table 2

shows the best assignment for soc h953 with different random seeds. The results for various SoC benchmarks are furnished in Table 3,4,5.

6. CONCLUSION

The method proposed here is to develop a scheduling scheme based on Genetic Algorithm so that effective scheduling is obtained and thereby getting an optimal solution by its inherent nature. The optimal solution thus generated is applied to the ITC'02 benchmark circuits and test application time is calculated for different TAM widths. The test time is also tabulated under the power constraints. As a future work, the GA proposed optimal test architecture can be added as a library module in the EDA tools used for SoC designing and testing. Hence from a testing perspective, an early Test Planning and Development is possible and thereby improving the overall SoC design flow from the early stage itself.

7. REFERENCES

[1] Zhan Jinyu, Li Xunsheng Guo Bing,XiongGuangze,Sang Nan, "A Test Scheduling Scheme for Core-Based SoC's using Genetic Algorithm ", International Conference on Embedded Software Systems Symposia,2008

[2] Saluja, K.K, Ramanathan, P. "Partition Based SoC Test Scheduling with Thermal and Power Constraints under Deep Submicron Technologies" , IEEE Xplore 2014

[3] Chandan Giri,Dilip Kumar Reddy and Santanu Chattopadhyay,"Genetic Algorithm Based Approach for Hierarchical SOC Test Scheduling", Proceedings of the International Conference on Computing: Theory and applications (ICCTA '07)

[4] Chandan Giri et al., "Genetic Algorithm based heuristic technique for power constrained test scheduling in Core based SoC," International conference on Very Large Scale Integration, (VLSI-SoC 2007).

[5] Erik Jan et. al. " A set of Benchmarks for modular Testing of SoCs", ITC International Test Conference,2002.

[6] Wei Zou et. al. "SOC Test Scheduling Using Simulated

[7] Annealing ", Proceedings of the 21st IEEE VLSI Test Symposium, 2003. Li Hong,Xiong Shibo "On Ant Colony Algorithm for Solving Continuous Optimization Problem ",Intelligent Information Hiding and Multimedia Signal Processing, 2008.

[8] P.sakthivel et. al. "Design and Optimization of Test solutions for Core based System on Chip Benchmark circuits using Genetic Algorithm",Proceedings of the World Congress on Engineering,2007

[9] K. Chakrabarty, "Design of system on chip test access architecture using linear programming," in Proc. 18th IEEE VLSI Symposium, 2000, pp. 127-134.

[10] V. Iyengar, K. Chakrabarty, and E. J. Marinissen, "Test wrapper and test access architecture co-optimization for system-on-chip,Journal of Electronic Testing Theory and Application, vol. 18, pp. 213-230, March 2002.

[11] E. Larsson and Z. Peng, "System-on-chip test parallelization under power constraints," in Proc. IEEE European Test Workshop,May 2001.

[12] C. P. Su and C. W. Wu, "A graph-based approach to power constrained SoC test scheduling," In JETTA, pp. 45-60, 2004.

[13] D. Zhao and S. Upadhyay, "A generic Resource distribution and test scheduling scheme for embedded core-based SoCs," IEEE Trans. Instrumentation and Measurement, vol. 53, pp. 318-329, April 2004.Krishnendu Chakrabarthy

[14] Test Scheduling for Core based Systems using Mixed Integer Linear Programming", IEEE Transactions on Computer aideddesign of Integrated circuits and systems,Vol 19,No.10,October2000.