Power Aware Network on Chip Test Scheduling with Variable Test Clock Frequency

Harikrishna Parmar(✉) and Usha Mehta

EC Department, Nirma University, Ahmedabad, Gujarat, India
hk_parmar@yahoo.co.in, usha.mehta@nirmauni.ac.in

Abstract. For a stated core, the test time changes in a staircase pattern with the width of Test Access Mechanism (TAM). The core test time cannot decrease all time with increase in TAM width. However, the test time can always be diminished with increasing the test clock speed but clock speed cannot be increased beyond power limits. Here, a new method is proposed to reduce the Network on Chip (NoC) test time, by differing the test clock frequency such that it doesn't cross the predefined power limit. The power dissipation, test clock frequency and overall test time is the three trade off. In the proposed method, the clock frequency is optimized to minimize the total test application time (TAT) considering the power limits. Experimental results show an reduction of 48% over existing solution for the benchmark system on chip (SoC) D695, P93791 and P22810.

Keywords: Power · NoC · TAM
Test clock frequency · Overall test application time (TAT)

1 Introduction

Now a days, huge number of transistors are integrated on a wafer, which shows the certainty of Gordan Moore prediction, who had stated that the number of transistor on a wafer will be doubled on every 18 months. State-of-the-art technologies for manufacturing integrated circuits allow integrating a huge number of transistors on a single chip and reuse of IP cores for the sake of settling the time-to-market issues [1]. However, as the SoC is becoming more and more complex, typical bus based TAM architecture for SoC is subjected to scalable global synchronous clock, performance issues and communication bandwidths [2]. To avoid the limitation of SoC bus based architecture, Network on chip (NoC) system is introduced. Routers, channels, IP Cores and packet switching interconnections make NoC systems ideal to overcome limitations of SoC. [3]. The NoC system dispenses multiple benefits over long-established bus based architecture for its superior parallelism.

NoC is an emerging design paradigm deliberated to cope with future systems-on-chips (SoCs) comprising numerous built-in cores. Since NoC have some excellent distinctive attribute like scalability, design complexity, power dissipation, timing and so on, extensive interest is probable to grow towards NoC. The test strategy is the main

N. Kumar and A. Thakre (Eds.): UBICNET 2017, LNICST 218, pp. 256–264, 2018.
https://doi.org/10.1007/978-3-319-73423-1_23

aspect in the feasibility and practicability of the NoC based SoC. In SoC, TAM architecture is designed to fetch the test data from the automatic test equipment (ATE) to the core and to transfer test response from core to sink. Among the existing test objective for NoC based SoC, test scheduling and TAM architecture peculiarly influence the overall test performance of NoC [4].

Since, minimization of test time with NoC as TAM is an intractable problem, requiring the co-optimization of the core assignment to TAM for test data transportation, effective exploitation of the channel bandwidth and the number and location of the test interface. One or more of these features were ignored in the past.

In this paper, a new method is proposed to minimize the test time in NoC by differing the test clock frequency for each test session. Here, the test power and the test time is formulated as a function of test clock frequency, and hence this method gets the test time reduction for the predefined power limit. In the proposed method dynamic clock control based on power dissipation of test session is adopted.

The paper is organized as follow: Sect. 2 covers the prior work similar to the NoC testing whereas Sect. 3 covers the proposed algorithm based on variable test clock rate. Outcomes are discussed in Sect. 4 and then Sect. 5 concludes the paper.

2 Prior Work

Prior work shows the test time as a function of the TAM width and assignment of core to the TAM width to minimize the test time [5]. TAM architecture is the mediator to transfer the test data from automatic test equipment to the core and core to the sink. In [6], it is shown that the test time of the core varies in staircase pattern with TAM width. [5, 7–9] shows the optimal assignment of TAM width to the core under test to reduce test time significantly.

In NoC, NoC fabric can be used as TAM. So, the requirement of dedicated TAM can be avoided here. Since no extra hardware is required to build TAM, it reduces the cost of NoC testing [10, 11]. Number of scheduling algorithm is designed to minimize NoC test time with different constraints are proposed in [12–20].

The fundamental of reusing NoC as TAM are first introduced in [4]. Here, the core having longer test time is given higher priority in scheduling to reduce testing time. This method was further developed in [10] with power constraint and increased test parallelism. The Time division multiplexing (TDM) approach was discussed in [18] to have high speed test data transportation over the network and low speed test execution of NoC core. In [16, 17], Poweraware test scheduling is shown by effectively utilizing on chip network. Here the on chip clocking is used in a smart way such that the faster clock is assigned to some cores and slower to remaining to limit the overall power consumption. In short, clock rate distribution is effectively designed in this methodology to have lower test time. Test scheduling using rectangle packing solution and use of multiple test clocks for NoC test was proposed in [21]. Test scheduling with different topology of network was described in [12]. It also gives the idea of fast wiring test time minimization blueprint for different test structure. In [22], a unicast based multicast complication of NoC core testing is explained, where different techniques like Test data compression, power constraint scheduling, vector compactions are used to minimize test

time. Power and thermal aware NoC test scheduling with multiple clock rate is proposed in [23]. The algorithm is designed based on Integer linear programming and simulated annealing technique. Co-optimization of pin assignment to access point and NoC core test scheduling was proposed in [24]. Minimization of test time with given pin count is well described here. In [25], test delivery optimization of many core system is proposed. Here, NoC partitioning difficulty is formulated with dynamic programming. It also emphasize the optimization of the access point location, distribution of automatic test equipment (ATE) to access point and assignment of core to access point. In [26], hybrid test data transportation system for advance NoC based SoC is described. As the scheduler is affected by the location of the access point and the position of the embedded core, a new technique is developed here for concurrently testing several diverse cores.

3 Proposed Method

Assume that there are 'n' numbers of cores $C_1 \ldots C_n$ in an NoC. Individual core is initialized with its test time t_i and power consumed P_i. Maximum power limit of the NoC is P_{max}. Assume that core can be scheduled individually or in a group called sessions. Each session can have more than one core. The length of each session can be defined as

$$L_{Sj} = \max(t_i | \text{for all } t_i \in S_j) \tag{1}$$

And the power dissipated in that session can be defined as

$$P_{Sj} = \sum (P_i), \text{for all } t_i \in S_j \tag{2}$$

Since, routine test scheduling algorithm doesn't give any information regarding test clock frequency, here assume that all test time and power are evaluated on nominal clock frequency f_{nom}. Since, frequency is inversely proportional to the overall test time along with directly proportional to the power, increase in the clock frequency decreases the test time but increase power. Keeping this fundamental in mind, a new idiom is introduced called Frequency factor 'F', which will decide, for how many fold - frequencies should be increased or decreased to have optimum test time of an NoC. To understand this fundamental, two cases are evaluated here.

Case 1: If each core is scheduled individually.

$$\begin{aligned} \text{Frequency factor} \quad = F_{core1} &= P_{max}/P_{core1} \\ F_{core2} &= P_{max}/P_{core2} \\ &\vdots \\ F_{coren} &= P_{max}/P_{coren} \end{aligned}$$

Case 2: If each core is scheduled in a group called session.

e.g. If core 1, 2 … m of n cores are scheduled in a group, then

$$\text{Frequency factor} = F_{session1} = P_{max}/(P_{core1} + P_{core2} \ldots P_{corem})$$

In both the case, if Frequency factor F is greater than 1 then the test clock frequency will be increased by frequency factor times and if Frequency factor F is less than 1 then the test clock frequency will be decreased by frequency factor times. If F = 1 then test clock frequency will remain unchanged which indicates that all the sessions are executed at the nominal frequency f_{nom}. Now, the test scheduling of sessions can be framed as

Objective: Min $\sum(L_{Sj}/F_j) * x_j$ for j = 1 to k

where $x_j = 1$, if S_j is Scheduled

0, otherwise

Constraints: (1) $P_{Sj} * F_j * x_j \leq P_{max}$, where P_{max} is the power limit for the NoC. (2) Each core, Ci, $i \in \{1, 2, \ldots, n\}$ is made performed at least once.

Here, the first constraint indicates that frequency factor F cannot be increased more than (P_{max}/P_{si}), so that power constraint will not be violated. Power of discrete core can be intensify up to P_{max} but not beyond it. The power limit and test time of SoC D695 is depicted in Table 1.

If each core is scheduled individually, then the lower bound of the test time is set, as the test-clock-frequency is increased till the power consumed for each core is the same as the power limit P_{max}. The results are represented in Table 2 for case II and case I results are shown in Tables 3 and 4 for SoC D695.

3.1 Test Time Calculation

The entire test time is set by

$$T = \max(1 \leq j \leq B) \sum(T_{testi}) \text{ for } i = 1 \text{ to } n \quad (3)$$

where, B is number of test session

T_{testi} is the test−time of all cores on TAMj

n is the overall number of core.

Here, T_{testi} is the combination of two entities 1. Transmit time T_{trai} 2. Test time of core T_{corei}

$$T_{testi} = T_{trai} + T_{corei} \quad (4)$$

Since, transmit time depends on number of channels and routers used in NoC. So, it can be given as,

$$T_{trai} = nb_{chan} * T_{chan} = nb_{ro} * T_{ro} \quad (5)$$

where T_{ro} is the time consumed in router

T_{chan} is the time consumed in NoC channels

nb_{chan} is number of channels

nb_{ro} is number of routers.

Core test time depends on the TAM width selection and arrangement of scan chain with Best Fit Decreasing algorithm [21]. So total core test time is given as

$$T_{corei} = (1 + \max(S_i, S_o)) * p + \min(S_i, S_o) \quad (6)$$

where S_i = Wrapper−scan−in chain

S_o = Wrapper−scan−out chain

P = Test pattern count of the core

In NoC, Testing time of core is considerably higher than the transmit time. So here, transmit time is neglected as in contrast to core test time. Here, S_i and S_o is basically flip flops and it works on the edge of clocks, so the test time measured here is in number of clock cycle it used.

4 Results and Discussions

Here, the proposed algorithm is implemented on three SoC D695, P93791 and P22810 form ITC 2002 benchmark [27]. Since power consumption of each core is not mentioned in ITC 2002 benchmark database, it is taken from [28] where power consumption is calculated from the number of input, output and scan chain. For the proposed algorithm, here it is assumed that NoC has similar configuration as given in [17, 25, 29] like network topology, core placement etc. In all SoC, each core has different combination of scan chain, input, output and circuit structure, so the power consumption varies from core to core. Since power $P = fCV^2$, we are keeping capacitance - C and voltage - V constant and analyzing the effect of changing the frequency on power.

Here, for power constraint test scheduling, maximum power limit is set as the percentage of the gross of total power consumption of sole core i.e. 30% power edge means 30% of summation of total power consumed by each core.

Test database for SoC D695 is shown in Table 1. Column 1 list the core numeral, Column 2 catalogue the test time in clock cycles when TAM width is equal to 32. Core test time is evaluated from Eq. 6. Column 3 lists the power consumption of each individual core.

Here, the simulation is done on MATLAB 14, and LPSOLVE. The results of the proposed algorithm is compared with [17, 25, 29] and shown in Table 5 with SoC D695, P22810 and P93791 respectively with different maximum power limit and different Input output. Column 2–3 shows the results generated from [17] with 2 cases (1) 50% power limits (2) 30% power limit. Results shown in Tables 3 and 4 are the

smallest test time achieved ever. Results are compared with [29] and it shows that for 50% power limit, average test time reduces by 48% and for 30% power limit; test time reduces by 24%.

Table 1. Test database for SoC D695

Core	Test clock cycles	Power
Core-1	25	600
Core-2	584	602
Core-3	2475	823
Core-4	5775	275
Core-5	5843	690
Core-6	9828	354
Core-7	3325	530
Core-8	4559	753
Core-9	834	640
Core-10	3859	1144

Table 2. Results with SoC D695 with 50% power limit (core scheduled in sessions)

Core	Power	Time	Frequency factor	Test time
Core-1, 2, 3, 4, 5	2990	5843	1.07	5460
Core-6, 7, 8, 9, 10	3421	9828	0.93	10567
			Total	16027

Table 3. Results with SoC D695 with 50% power limit (core scheduled individually)

Core	Power	Time	Frequency factor	Test time
Core-1	600	25	6.35	4
Core-2	602	584	5.33	110
Core-3	823	2475	3.9	635
Core-4	275	5775	11.067	522
Core-5	690	5843	4.65	1257
Core-6	354	9828	9.06	1085
Core-7	530	3325	6.05	550
Core-8	753	4559	4.26	1070
`Core-9	640	834	5.0	167
Core-10	1144	3859	2.80	1378
			Total	6778

Table 4. Results with Soc D695 with 30% power limit (core scheduled individually)

Core	Power	Time	Frequency factor	Test time
Core-1	600	25	3.2	8
Core-2	602	584	3.19	183
Core-3	823	2475	2.33	353
Core-4	275	5775	7.0	825
Core-5	690	5843	2.78	2101
Core-6	354	9828	5.43	1810
Core-7	530	3325	3.62	919
Core-8	753	4559	2.55	1788
Core-9	640	834	3.00	278
Core-10	1144	3859	1.68	2297
			Total	10562

Table 5. Results with SoC D695, P93791, P22810 with I/O = 2/2

SoC	From ref [29]		Proposed			
	50%	30%	50%	Reduction	30%	Reduction
D695	11927	14250	6778	44%	10562	25%
P93791	443548	444350	203117.3	54%	338534.6	23%
P22810	165302	165302	8447	48%	12389	24%
			Average reduction	48%	Average reduction	24%

5 Conclusion

Here, it is proved that significant test time minimization is achieved by managing the test clock frequency of the test sessions. For the given assumption that the clock frequency confined through the power limit of the NoC, optimization attained is much better. It's also shown that, if the cores are scheduled individually, then the optimum test time is achieved, thereby setting the lower bound of the test time in NoC. Experimental results present an enhancement of 48% on to existing solution for the benchmark SoC D695, P93791 and P22810.

References

1. Moreno, E., Webber, T., Marcon, C., Moraes, F., Calazans, N.: NoC: a monitored network on chip with path adaptation mechanism. Syst. Archit. **60**, 783–795 (2014)
2. Ansari, A., Song, J., Kim, M., Park, S.: Parallel test method for NoC-based SoCs. In: Proceedings IEEE International SoC Design Conference (ISOCC), pp. 116–119 (2009)
3. Touzene, A.: On all-to-all broadcast in dense Gaussian network on-chip. IEEE Trans. Parallel Distrib. Syst. **26**, 1085–1095 (2015)

4. Cota, E.: The impact of NoC reuse on the testing of core-based systems. In: Proceedings of the IEEE VLSI Test Symp, pp. 128–133 (2003)
5. Larsson, E.: Introduction to Advanced System-on-Chip Test Design and Optimization. Springer, Heidelberg (2005). https://doi.org/10.1007/b135763
6. Iyengar, V., Chakrabarty, K., Marinissen, E.: Test wrapper and test access mechanism co-optimization for system-on-chip. J. Electron. Test. Theory Appl. **18**, 213–230 (2002)
7. Iyengar, V., Chakrabarty, K., Marinissen, E.J.: On using rectangle packing for SOC wrapper/TAM co-optimization. In: Proceedings of the 20th IEEE VLSI Test Symposium (2002)
8. Zhao, D., Upadhyaya, S.: Power constrained test scheduling with dynamically varied TAM. In: Proceedings of the 21st IEEE VLSI Test Symposium (2003)
9. Larsson, E., Fujiwara, H.: Power constrained preemptive TAM scheduling. In: Proceedings of the Seventh IEEE European Test Workshop (2002)
10. Cota, E., Carro, L., Lubaszewski, M.: Reusing an on-chip network for the test of core-based systems. ACM Trans. Des. Autom. Electron. Syst. **9**, 471–499 (2004)
11. Cota, E., Liu, C.: Constraint-driven test scheduling for NoC based systems. IEEE Trans. Comput. Aided Des. Integr. Circ. Syst. **25**, 2465–2478 (2006)
12. Amory, A., Lazzari, C., Lubaszewski, S., Moraes, S.: A new test scheduling algorithm based on networks-on-chip as test access mechanisms. J. Parallel Distrib. Comput. **71**, 675–686 (2011)
13. Chakrabarty, K.: Test scheduling for core based systems using mixed-integer linear programming. IEEE Trans. Comput. Aided Des. Integr. Circuits Syst. **19**, 1163–1174 (2000)
14. Chattopadhyay, S., Reddy, K.: Genetic algorithm based test scheduling and test access mechanism design for system-on-chips. In: Proceedings of the 16th International Conference on VLSI Design, pp. 341–346 (2003)
15. Iyengar, V., Chakrabarty, K.: System on-a-chip test scheduling with precedence relationships, preemption, and power constraints. IEEE Trans. Comput. Aided Des. Integr. Circuits Syst. **21**, 1088–1094 (2002)
16. Liu, C., Iyengar, V.: Test scheduling with thermal optimization for network-on-chip systems using variable-rate on-chip clocking. In: Proceedings Design, Automation and Test in Europe Conference and Exhibition (DATE), pp. 650–655 (2006)
17. Liu, C., Shi, J., Cota, E., Iyengar, V.: Power-aware test scheduling in network-on-chip using variable-rate on-chip clocking. In: Proceedings of the 23rd IEEE VLSI Test Symposium (VTS), pp. 349–354 (2005)
18. Nolen, M., Mahapatra, R.: TDM test scheduling method for network-on-chip systems. In: Proceedings of the Sixth International Workshop on Microprocessor Test and Verification (MTV), pp. 90–98 (2005)
19. Su, C., Wu, C.: A graph-based approach to power-constrained SOC test scheduling. J. Electron. Test. Theory Appl. **20**, 45–60 (2004)
20. Zou, W., Reddy, M., Pomeranz, I., Huang, Y.: SOC test scheduling using simulated annealing. In: Proceedings of the 21st IEEE VLSI Test Symposium (VTS), pp. 325–330 (2003)
21. Ahn, J., Sungho, K.: Test scheduling of NoC-based SoCs using multiple test clocks. ETRI J. **28**, 475–485 (2006)
22. Xiang, D., Zhang, Y.: Cost-effective power-aware core testing in NoCs based on a New unicast-based multicast scheme. IEEE Trans. Comput. Aided Des. Integr. Circuits Syst. **30**, 135–147 (2011)
23. Aktouf, A.: Complete strategy for testing an on-chip multiprocessor architecture. IEEE Des. Test Comput. **19**, 18–28 (2002)

24. Richter, M., Chakrabarty, K.: Optimization of test pin- count, test scheduling, and test access for NoC-based multicore SoCs. IEEE Trans. Comput. **63**, 691–702 (2014)
25. Agrawal, M., Richter, M., Chakrabarty, K.: Test-delivery optimization in manycore SOC. IEEE Trans. Comput.-Aided Des. Integr. Circuits. Syst. 33(7) (2014)
26. Ansari, A., Kim, D., Jung, J., Park, S.: Hybrid test data transportation scheme for advanced NoC-based SoCs. J. Semicond. Technol. Sci. **15**, 85–95 (2015)
27. Marinissen, E., Iyengar, V., Chakrabarty, K.: A set of benchmarks for modular testing of SOCs. In: Proceedings International Test Conference (ITC), pp. 519–528 (2002)
28. Pouget, J., Larsson, E., Peng, Z.: SOC test time minimization under multiple constraints. In: Proceedings of the 12th Asian Test Symposium (ATS), pp. 312–317 (2003)
29. Hu, C., Li, Z., Lu, C., Jia, M.: Test scheduling for network-on-chip using XY-direction connected subgraph partition and multiple test clocks. J. Electron. Test. **32**, 31–42 (2016)