Test Scheduling for Wafer-Level Test-During-Burn-In of Core-Based SoCs

Sudarshan Bahukudumbi and Krishnendu Chakrabarty
Department of Electrical and Computer Engineering
Duke University, Durham, NC 27708
Email: {spb, krish}@ee.duke.edu

Richard Kacprowicz
Intel Corporation
Hillsboro, OR 97124
Email: richard.kacprowicz@intel.com

***Abstract*—Wafer-level test during burn-in (WLTBI) has recently emerged as a promising technique to reduce test and burn-in costs in semiconductor manufacturing. However, the testing of multiple cores of a system-on-chip (SoC) in parallel during WLTBI leads to constantly-varying device power during the duration of the test. This power variation adversely affects predictions of temperature and the time required for burn-in. We present a test-scheduling technique for WLTBI of core-based SoCs, where the primary objective is to minimize the variation in power consumption during test. A secondary objective is to minimize the test application time. Simulation results are presented for two ITC'02 SoC benchmarks, and the proposed technique is compared with two baseline methods.**

I. Introduction

System-on-chip (SoC) integrated circuits pose a number of challenges for manufacturing test [1]. In addition to the need for effective test techniques for defect screening and speed binning, there is an ever-increasing demand for high device reliability and low defect-per-million levels. Semiconductor manufacturers routinely perform reliability screening on all devices before shipping them to customers [2]. Accelerated test techniques shorten the time-to-failure process without altering the device failure characteristics [3]. Burn-in is one such technique that is widely used in the semiconductor industry [3], [4].

The long time intervals associated with burn-in often result in high cost [1], [5]. Wafer level burn-in (WLBI) has recently emerged as an enabling technology to lower the cost of burn-in [4]. In this approach, devices are subjected to burn-in and electrical testing while in the bare wafer form. By moving the burn-in process to the wafer-level, significant cost savings can be achieved in the form of lower packaging costs, and reduced burn-in and test time. WLBI is performed in a massively parallel manner across the wafer [4]; this contributes further in maximizing cost savings.

Test during burn-in at the wafer-level enhances the benefits that are derived from the burn-in process. The monitoring of device responses while applying suitable test stimuli during WLBI leads to the easier identification of faulty devices. We refer to this process as "wafer-level test-during-burn-in" (WLTBI); it is also referred to as "test in burn-in" (TIBI) [3], "wafer-level burn-in test" (WLBT) [6], etc. WLTBI technology has recently made rapid advances with the advent of the "known good die" (KGD) [7], i.e. devices that are sold as tested bare die. In the manufacture of KGDs, WLTBI eliminates the need for a die-carrier and carrier burn-in, thereby resulting in cost savings.

WLTBI can lower product cost by breaking the barrier between burn-in and test processes. Automatic test equipment (ATE) manufacturers have introduced WLBI and test equipment that provide full-wafer contact during burn-in and they also provide test monitoring capabilities [4], [6]. Each device receives the same test stimulus during WLTBI. Monitoring the device responses during burn-in are a key part of test during burn-in strategies.

Concurrent testing of core-based SoCs reduces test time by testing multiple cores in parallel [8]. However, it also results in fluctuating power consumption of the device over the period of test application. This variation in test power results in frequently varying junction temperatures of the SoC. Modeling the time required for burn-in takes into account a fixed value of power for the entire device [9], [10]. The testing of a core-based SoC in a burn-in environment will therefore adversely affect predictions on burn-in time (resulting in a device being subjected to excessive or insufficient burn-in), and in certain cases may result in thermal runaway.

In this paper, we present a power-conscious test-scheduling technique for WLTBI of core-based SoCs. This technique allows us to select cores that are tested in parallel while minimizing the overall variation in power. Maintaining the spread in power consumption during test will significantly lower the variations in junction temperature [9].

II. Thermal Challenges during WLTBI

The challenges that are encountered during WLTBI are a combination of the problems faced during the sort process and during burn-in. Current wafer probers use a thermal chuck to control the device temperature during the sort process. The chuck is an actively regulated metal device controlled by external chillers and heaters embedded under the device [9]. The junction temperature of the DUT is determined by the following relationship [9], [10]:

$$T_j = T_a + P \cdot \theta_{ja} \quad (1)$$

where T_j is the junction temperature of the device, T_a is the ambient temperature, P is the device power consumption, and θ_{ja} is the thermal resistance (junction to ambient) of the device. The value of T_j is therefore determined by the device power consumption, thermal resistance, and a constant T_a. The controllability of T_j is limited by the extent to which the parameters T_a and P can be controlled. Considerable power fluctuations during the test of the device under test (DUT) can significantly affect the value of T_j for the DUT, thereby adversely impacting the reliability screening process.

978-3-9810801-3-1/DATE08 © 2008 EDAA

One of the important goals of the burn-in process is to keep the burn-in time to a minimum, thereby increasing throughput, and minimizing equipment and processing costs. It is also important to have a tight spread in temperature distribution of the device to increase yield and at the same time minimize burn-in time [9]. The parameter T_j cannot exceed a pre-determined threshold due to concerns of thermal runaway and the need to maintain proper circuit functionality. It is this issue of controlling the spread in T_j over the period of test application that we address in this paper. The problem of controlling the power profiles, which depend on the test schedule, has been ignored thus far in literature. We therefore develop a power-conscious test scheduling approach, specifically suited for WLTBI of core-based SoCs.

III. Test scheduling for WLTBI

Efficient test-scheduling methods such as [8], target increased test concurrency to reduce the test application time. This leads to increased power consumption during test. Recent test-scheduling techniques for core based SoCs have included the additional dimension of test power consumption [11], [12]; this ensures that a pre-determined limit on power consumption is not exceeded during test. These techniques, however, do not address the variations in power that occur during test application. We develop a power-conscious test scheduling approach in this paper, tailored for WLTBI of core-based SoCs. The primary objective of our work is to minimize variations in power consumption such that predictions on burn-in time are accurate. A secondary objective is to minimize the test application time.

A. Core-ordering problem for WLTBI

We assume a fixed-width TAM architecture and test buses [8], where the division of W wires into B TAM partitions has been determined *a priori* using methods described in [8]. We now have to determine an optimal ordering of cores such that the overall variation in power consumption for the SoC is minimized while satisfying the constraint on peak power consumption P_{max}. We refer to this problem as $\mathcal{P}_{Core\_Order}$. We use the following two measures as metrics to analyze the variation in power consumption.

1) The first measure is the statistical variance in test power consumption. Let T_{SoC} represent the test time for the SoC in clock cycles, and P_{mean} the mean value of power consumption per clock cycle during test. The variance in test power consumption for the SoC is defined as $\frac{1}{T_{SoC}} \sum_{i=1}^{T_{SoC}} (P_i - P_{mean})^2$. Low variance indicates low (aggregated) deviation in test power from the mean value of power consumption during test. Successful WLTBI requires the minimization of this metric.
2) The cycle-to-cycle variation in test power consumption is an indicator of the "flatness" of the power profile during test. Large cycle-to-cycle power variations are undesirable. We therefore quantify the "flatness" in the power profile using the metric $\gamma = \frac{\sum_{i=1}^{T_{SoC}-1} |P_{i+1} - P_i|}{T_{SoC}-1}$; P_i and P_{i+1}, denote the power consumption during the i^{th} and $(i+1)^{th}$ clock cycles. Low values of γ are desirable for WLTBI.

Without loss of generality and to simplify the presentation, we henceforth consider an SoC with three TAM partitions ($B = 3$). (The extension to more than three TAM partitions is straightforward.) The problem $\mathcal{P}_{Core\_Order}$ for an SoC with three TAM partitions can now be formally stated as follows:

Problem $\mathcal{P}_{Core\_Order}$: Let T_1, T_2 and T_3 be the sets of cores on TAM partitions 1, 2 and 3 respectively. Determine the sets of cores that can be tested simultaneously, and the ordering of the cores on the TAM partitions, such that the overall variation in power consumption for the SoC is minimized and the peak power constraint P_{max} is satisfied.

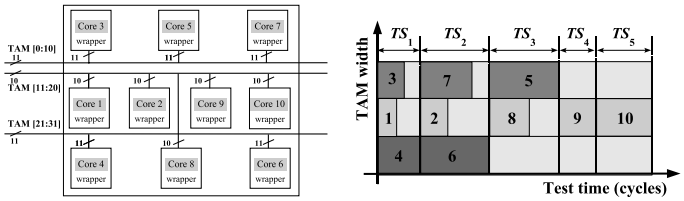
We use the parameter $\rho(i, j, k)$ to represent the variation in power consumption when the three cores i, j, and k are tested in parallel. It is given by $\rho(i, j, k) = \mu(i, j, k) + \sigma(i, j, k)$; the parameter $\mu(i, j, k)$ is the statistical mean and $\sigma(i, j, k)$ is the standard deviation in power consumption, when cores i, j, and k are tested concurrently. Note that $1 \leq i \leq |T_1|$, $1 \leq j \leq |T_2|$ and $1 \leq k \leq |T_3|$.

B. Heuristic procedure to solve $\mathcal{P}_{Core\_Order}$

We next describe the heuristic algorithm that we use to solve $\mathcal{P}_{Core\_Order}$. The algorithm starts with an initial assignment of cores to TAM partitions, and then iteratively (re)assigns cores to the three TAM partitions such that the variation in test power is minimized. Every step in the heuristic also ensures that the power constraint P_{max} is satisfied. The main steps in the heuristic are outlined below:

1) In procedure $Initial\_Assign$, we schedule cores that are tested first on each TAM partition, i.e., their test start-times are zero. The assignment of cores is obtained by determining the triple that yields the lowest value for $\rho(i, j, k)$. A triple corresponds to a set of three cores, one from each TAM partition.
2) In procedure $Assign\_Cores$, we determine the next sets of cores that are assigned to the test schedule. Cores are iteratively scheduled in sets of three, until there are no more valid triples.
3) In procedure $Unmatched\_Assign$, we determine the assignment of cores (vertices) that have not been scheduled. If all the cores in a particular TAM partition have already been scheduled, $Unmatched\_Assign$ selects cores from the remaining TAM partitions to reduce the overall variation in test power.

An example of a TAM architecture for the d695 SoC with a TAM width $W = 32$ is shown in Figure 1(a). Figure 1(b) illustrates the corresponding test schedule obtained using the heuristic method. The first two test sessions TS_1 and TS_2 in the test schedule correspond to sets of cores $\{3, 1, 4\}$, and $\{7, 2, 6\}$ that are tested simultaneously. Cores 3, 1, and 4 when tested concurrently result in the least power variation among all valid core combinations. The power data for this example is taken from the cycle-accurate test modeling approach presented in [11]. The two cores $\{5, 8\}$ are tested during test



(a) (b)

Fig. 1. (a) TAM architecture for the d695 SoC with $W = 32$. (b) Test schedule for the d695 SoC with $W = 32$ and $P_{max} = 1800$.

session TS_3 as shown in Figure 1(b). Finally, cores 9 and 10 are tested individually in the test schedule.

The proposed heuristic solution can be easily extended for SoCs with more than three TAM partitions. The $Initial\_Assign$ procedure and the $Assign\_Cores$ procedure both require searching through N^3 candidate solutions in the worst case; hence the time complexity is $O(N^3)$, where N is the number of cores in the SoC. The worst-case time complexity of the heuristic procedure in terms of the number of TAM partitions B is $O(N^B)$. The heuristic procedure is exponential in the number of TAM partitions B, but B is a constant at wafer-level since the TAM architecture is optimized during design time for package test.

We next describe two baseline methods used in this paper. The first baseline method solves a power-constrained test-scheduling problem for core-based SoCs. This approach considers a single power-limit value for the entire SoC [11]. We determine the variation in power consumption over time, when only a peak power limit is considered for test scheduling. We use the same TAM architecture used by the $Core\_Order$ heuristic.

The baseline scheduling algorithm keeps a record of the per-cycle values of power consumption and ensures that it is less than P_{max} at every cycle. When a new core is added to the test schedule, the test power for the core is accumulated to reflect the overall power consumption profile of the SoC. The algorithm iteratively schedules the cores in the SoC to minimize the SOC test time, while satisfying the power limit P_{max}.

In the second baseline method, we consider a pre-designed TAM architecture, where the division of W top-level TAM wires into B TAM partitions, and the assignment of cores to these TAM partitions are determined *a priori* using methods described in [8] for package test. We then test these cores serially with their pre-allocated TAM width, such that the power consumption and the variance in power consumption are kept to a minimum. No two cores are tested concurrently.

IV. Experimental Results

In this section, we present experimental results for three SoCs from the ITC'02 SoC test benchmarks. We use cycle-accurate power data from [11]. Since the objective of $\mathcal{P}_{Core\_Order}$ is to minimize the variation in test power consumption (represented by the two metrics presented in Section III) during WLTBI, we present the following results:

- The percentage difference in variance between baseline method 1 and $Core\_Order$. This difference is denoted by $\delta V_{Baseline1}$, and it is computed as $\frac{V_{Baseline1} - V_{Core\_Order}}{V_{Baseline1}} \times 100\%$; V_{Core_Order} represents the variance in test power consumption obtained using the $Core\_Order$ heuristic, and $V_{Baseline1}$ represents the variance in power consumption obtained using the first baseline method.

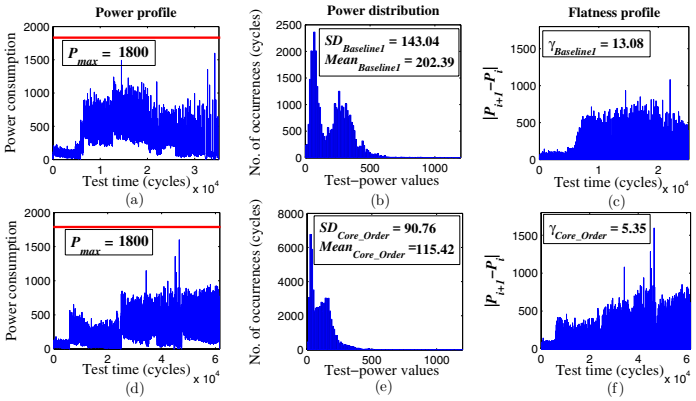


Fig. 2. Power profile for d695 obtained using baseline approach 1 and $Core\_Order$ ($W = 32$ and $P_{max} = 1800$).

- The percentage difference in variance between baseline method 2 and $Core\_Order$. This is calculated in a similar fashion as $\delta V_{Baseline1}$, and is denoted as $\delta V_{Baseline2}$.
- We highlight the difference in the mean cycle-to-cycle power variation obtained using baseline method 1, and $Core\_Order$. We characterize this difference as $\delta\gamma = \frac{\gamma_{Baseline1} - \gamma_{Core\_Order}}{\gamma_{Baseline1}} \times 100\%$; $\gamma_{Baseline1}$ and γ_{Core_Order} are the "flatness" indicators obtained using the first baseline method and the $Core\_Order$ heuristic respectively.
- We also present the WLTBI test time for the SoC obtained using $Core\_Order$ and the baseline test methods.

We first present power profiles and the corresponding distribution in power consumption values during test. Figure 2 illustrates the power profile for the d695 SoC when tested with a TAM width of 32; the maximum value of power consumption, P_{max}, is set to 1800 units in this case. (The units are derived from [11].) Figures 2(a) and 2(b) represent the power profile during test for the baseline approach and the distribution in power consumption values corresponding to the power profile, respectively; Figures 2(d) and 2(e) represent the same information obtained using the $Core\_Order$ heuristic. Figures 2(c) and 2(f) illustrate the flatness profiles obtained for the baseline scenario and using $Core\_Order$ respectively. We can make the following observations from Figure 2:

- The standard deviation SD, and hence the variance in power during test, is significantly lower when $Core\_Order$ is used to determine the ordering of cores.
- The mean value of power consumption (Mean) during test is also significantly lower when the cores are ordered using $Core\_Order$. This is because $Core\_Order$

reduces the variation in power consumption at the cost of increased test time.

- The lower values of variance in power consumption obtained using the $Core\_Order$ heuristic results in a distribution where the power consumption values are packed into fewer bins in the power distribution profile as compared to the baseline approach.
- The power profile obtained using $Core\_Order$, for the case illustrated in Figure 2, is 59% flatter than the baseline scenario. This is an indicator of the low cycle-to-cycle power variation during test.

P_{max}	W	$\delta V_{Baseline1}$	$\delta V_{Baseline2}$	$\delta\gamma$	TT_{Core_Order} (cycles)	$TT_{Baseline1}$ (cycles)	$TT_{Baseline2}$ (cycles)
	16	65.49	−13.37	49.61	124402	60482	147568
	32	59.74	−7.71	40.95	65870	53833	77113
	40	26.55	−23.29	20.37	61589	47442	75283
	56	20.81	−25.49	25.02	42350	22569	49620
1600	64	12.57	−25.34	26.66	41882	21595	48740
	16	56.52	−21.38	48.12	120468	60481	147568
	32	59.74	−7.71	40.95	65870	53833	77113
	40	14.11	−21.01	5.76	61589	35124	75283
	56	7.46	−16.38	43.10	34860	22423	49620
2000	64	4.17	−15.74	43.32	32499	18726	48740

TABLE I
REDUCTION IN TEST-POWER VARIANCE FOR D695.

P_{max}	W	$\delta V_{Baseline1}$	$\delta V_{Baseline2}$	$\delta\gamma$	TT_{Core_Order} (cycles)	$TT_{Baseline1}$ (cycles)	$TT_{Baseline2}$ (cycles)
	16	43.38	−0.05	6.27	4937767	1890881	5851966
	32	31.28	−0.02	27.99	2272156	1427138	2860859
	40	25.64	11.24	28.55	1600355	1152953	2017488
	56	35.10	−0.84	39.95	1185860	736604	1613826
15000	64	17.49	−2.49	40.01	1045983	694142	1478334
	16	43.38	−0.05	10.02	4937767	1890881	5851966
	32	31.28	−0.02	27.99	2272156	1427138	2860859
	40	25.64	11.24	28.55	1600355	1152953	2017488
	56	35.10	−0.84	39.95	1185860	736604	1613826
20000	64	17.49	−2.49	40.01	1045983	694142	1478334

TABLE II
REDUCTION IN TEST-POWER VARIANCE FOR P93791.

The results for the two benchmark SoCs, d695 and p93791 are summarized in Tables I-II respectively; five different values of W are considered in each case. The values of P_{max} for each circuit are chosen carefully after analyzing the per-cycle test-power data provided in [11]. The minimum value of P_{max} is chosen such that a feasible schedule can be formulated using the given value of P_{max}. The SoC test time, TT_{Core_Order}, obtained using $Core\_Order$, and the SoC test time using the baseline cases, $TT_{Baseline1}$ and $TT_{Baseline2}$ are reported in addition to $\delta V_{Baseline1}$, $\delta V_{Baseline2}$, and $\delta\gamma$. The results show that significant reduction in test power variation can be obtained using our heuristic procedure, which ideally is the goal for WLTBI. Significant reduction in cycle-to-cycle power variation is observed for all scenarios when $Core\_Order$ is used to order the cores.

The test times for the proposed approach are higher than that for baseline method 1. Recall that test-time minimization is a secondary objective for WLTBI. The primary objective here is to minimize the test-power variance. Note that a limited increase in the test time is not a serious drawback because the wafer is subjected to relatively long intervals of burn-in.

The second baseline approach results in low values of variance for power consumption. This because the cores are tested sequentially in this case, thereby resulting in much higher test times as compared to the first baseline approach and $Core\_Order$. Higher test times result in higher memory requirements; this limits the number of die that can tested in parallel during WLTBI. Temperature and voltage cycling during burn-in result in the die being tested at different operating temperatures and voltages [13]. A reasonable test time is therefore necessary to support test repetitions under such a scenario. The tester scan clock frequency for the burn-in ATE is lower than that for a conventional ATE [13]. The significantly higher test time for the second baseline method renders the method unsuitable for WLTBI.

V. Conclusions

We have formulated a test-scheduling problem for WLTBI of core-based SoCs, which minimizes the variation in test power during test application. This is the first attempt to develop a test-scheduling solution to address thermal issues that arise during WLTBI. We have used cycle-accurate test-power data for the cores to solve the test-scheduling problem. We have presented a heuristic technique to solve $\mathcal{P}_{Core\_Order}$. Results for two ITC'02 SoC test benchmarks show that a significant reduction in power variation is obtained using the proposed method.

References

[1] Int. Technology Roadmap for Semiconductors, 2005 http://www.itrs.net/Common/2005ITRS/Home2005.htm.

[2] L. Yan and J. R. English, "Economic cost modeling of environmental-stress-screening and burn-in," *IEEE Trans. Reliability*, vol. 46, pp. 275–282, Jun. 1997.

[3] P. C. Maxwell, "Wafer-package test mix for optimal defect detection and test time savings," *IEEE Design & Test of Computers*, vol. 20, pp. 84–89, Sep. 2003.

[4] "A comparison of wafer level burn-in & test platforms for device qualification and known good die (KGD) production", http://www.delta-v.com/images/White_Paper_-_Comparing_WLBT_Platforms.pdf.

[5] M. F. Zakaria et al., "Reducing burn-in time through high-voltage stress test and Weibull statistical analysis," *IEEE Design & Test of Computers*, vol. 23, pp. 88–98, Sep. 2006.

[6] I. Y. Khandros and D. V. Pedersen, *Wafer-level burn-in and test*. U. S. Patent Office, May 2000, Patent number 6,064,213.

[7] A. Singh, P. Nigh, and C. M. Krishna, "Screening for known good die (KGD) based on defect clustering: an experimental study," in *Proc. Int. Test Conf.*, 1997, pp. 362–371.

[8] V. Iyengar, K. Chakrabarty, and E. J. Marinissen, "Test wrapper and test access mechanism co-optimization for system-on-chip," *Journal of Electronic Testing: Theory and Applications*, vol. 18, pp. 213–230, Apr. 2002.

[9] P. Tadayon, "Thermal challenges during microprocessor testing," *Intel Technology Journal*, pp. 1–8, 2000.

[10] A. Vassighi, O. Semenov, and M. Sachdev, "Thermal runaway avoidance during burn-in," in *Proc. Int. Reliability Physics Symposium*, 2004, pp. 655–656.

[11] S. Samii et al., "Cycle-accurate test power modeling and its application to SOC test scheduling," in *Proc. Int. Test Conf.*, 2006.

[12] V. Iyengar and K. Chakrabarty, "System-on-a-chip test scheduling with precedence relationships, preemption, and power constraints," *IEEE Trans. CAD*, vol. 21, pp. 1088–1094, Sep. 2002.

[13] "Innovative burn-in testing for SoC devices with high power dissipation", http://www.advantest.de/dasat/index.php?cid=100363&conid=101096&sid=17d2c133fab7783a035471392fd60862.