BIST Power Reduction Using Scan-Chain Disable in the Cell Processor

Christian Zoellin and Hans-Joachim Wunderlich
Institut für Technische Informatik
Universität Stuttgart
Pfaffenwaldring 47, D-70569 Stuttgart, Germany
email: {zoellin, wu}@informatik.uni-stuttgart.de

Nicolas Maeding and Jens Leenstra
IBM Deutschland Entwicklung
Schoenaicherstr. 220, D-71032 Böblingen, Germany
email: {nmaeding, leenstra}@de.ibm.com

***Abstract*— Built-in self test is a major part of the manufacturing test procedure for the Cell Processor. However, pseudo random patterns cause a high switching activity which is not effectively reduced by standard low power design techniques. If special care is not taken, the scan-speed may have to be reduced significantly, thus extending test time and costs.**

In this paper, we describe a test power reduction method for logic BIST which uses test scheduling, planning and scan-gating. In LBIST, effective patterns that detect additional faults are very scarce after a few dozens of scan cycles and often less than one pattern in a hundred detects new faults. In most cases, such an effective pattern requires only a reduced set of the available scan chains to detect the fault and all don't-care scan chains can be disabled, therefore significantly reducing test power.

***Index Terms*— Microprocessor test, BIST, low power test.**

I. Introduction

Modern processors structured like the Cell processor work with high operation frequency, occupy several hundreds of square-millimeters and show significant power consumption already during functional mode. Altogether, this puts a significant challenge on fault coverage, test application time and power consumption. The Cell processor implements a scan-based BIST strategy according to the STUMPS-architecture [1], [2] (see Figure 1). Average power consumption during test and especially during BIST is orders of magnitude higher than power consumption in functional mode [3]. This fact has lead to extensive research in recent years for minimizing power consumption during test.

For the power evaluation of a BIST environment, three metrics are most im- portant:

- The average power, which is the quotient of energy and test application time, and which is relevant for hot spots and reliability problems.
- The peak power, which corresponds to the maximal power consumed in the CUT at a clock cycle. If in test mode the number of switching devices is higher than in system mode, the correct function of the circuit is not guaranteed due to IR-drop and noise [4]. Even correct circuits may fail the test and yield may be reduced significantly.
- The consumed energy.

The flexible STUMPS architecture in the CELL processor can easiliy be used to switch off scan chains for reducing peak power. In this paper we present an efficient test scheduling and test planning method for reducing average power and energy as well.

Methods for power optimization during BIST in scan-based designs include:

- Toggle suppression of scannable flip-flops during shifting. The largest amount of power is lost during shifting, not just in the scan paths but in the combinational logic driven by the scannable flip-flops [5]. The shifting activity may be blocked by integrating some masking logic into the scan flip-flops [5], [6], or by using a 3-latch design where the states are held in one latch during shifting [7].
- Filtering off patterns not contributing to fault coverage. Pseudo-random patterns loose efficiency in the course of testing, and after a while the pattern generator creates mostly useless patterns. Significant savings are obtained if the shift clock is gated and useless patterns are not shifted in [5], [8], [9].
- Scan path segmentation and bypassing.
  Shifting activity can be further reduced by segmenting the scan paths and implementing bypasses [10], [11].
- Flipflop reordering in the scan path for reducing the

shifting activity [12], [13], [14].
- Circuit partitioning in order to block the propagation of switching events [15].
- Low power pattern generators.
  Hardware pattern generators are employed that generate power aware test sequences [16], [17], [18].
- In addition, clock gating techniques as used in system design may be extended to DfT techniques [19], [20].

The list of techniques mentioned above may not be complete, and most of them work quite well for benchmarks in a research environment. However, they are not directly applicable to industrial multi-million gate designs due to various practical constraints.

- Impact on system functionality, especially speed and timing in functional mode.
- Impact on test time and fault coverage.
- Complexity of required design steps, and compliance with the design flow.
- Lack of CAD support.

These constraints are taken into account by commonplace industrial solutions which include:

- Oversizing power supply, package and cooling to withstand the increased current during test.
- Testing with reduced operating frequency at the cost of higher test time and possibly lower fault coverage because of dynamic faults that are not detected.
- Partitioning of the chip and appropriate scheduling of the test for each partition.

All of these approaches essentially result in a higher test cost, either by means of more expensive equipment or higher test time.

When creating a test procedure for a high volume product [21] the main objective is to have high test coverage and a very small test time. An optimized test should also take into account the heat capacity of the bare die on the wafer. Although the heat capacity of the die is very small, the test time of the current test for the Cell Processor (26 milliseconds [21]) is in a range where this property can be successfully exploited. Therefore, a test power reduction technique has to be flexible enough to adapt to any given power envelope that may depend on physical and mechanical considerations as well as the desired test length.

The goal of the work presented here, was to develop a power-aware test strategy for the Cell processor under the constraints listed below:

1) Significant reduction of both average power and energy compared with the unaltered test strategy.
2) Keeping the identical fault coverage without increasing the test application time.
3) Flexible control of the power consumption over test time in order to adapt test power to the die's temperature.
4) No additional hardware changes on top of the design for test logic already integrated in the Cell processor.
5) Implementing the strategy by following the standard design flow and using the available computer aided design and test tools.

These goals could be obtained by exploiting the specific design for test features of the Cell processor which allow the individual control of the clocks of each of the scan chains. Hence, a new test scheduling method could be applied, enabling at each time just a subset of scan chains without sacrificing fault coverage and test time.

The rest of the paper is organized as follows. Section 2 describes the Cell processor, its functional and test characteristics and the LBIST design for test. Section 3 describes the test planning algorithm, which generates multiple test blocks. Experimental results in section 4 which evaluate different test schedules show the efficiency, potential and limitations of the method, which is evaluated with respect to the goals and limits.

II. The Cell Processor

The test methodology presented in this paper has been developed for the Cell processor whose clock-gating and DFT represent the challenges in the test of such industrial designs. It has been designed to provide very high computing power for (among others) multimedia applications, while being highly power efficient and therefore suitable for embedded systems.

A. Functional characteristics

The Cell processor is a multi-core system-on-a-chip with multiple micro-architectures on a single die (Figure 2). The current implementation of the Cell processor [22] consists of 250 million transistors on a $235mm^2$ die. It incorporates a PowerPC core with 2 Threads and 8 Synergistic Processing Elements (SPE, Figure 3) with 256kB local store SRAM each. The theoretical single precision floating point throughput for the Cell is >200GFlops at 3.2GHz. The system of 9 cores has up to 25.6GB/s memory bandwidth and up to 50+GB/s I/O bandwidth available.

The whole chip consists of 15 LBIST domains [21]. Because 70% of the chip area is covered by the 8 identical SPEs

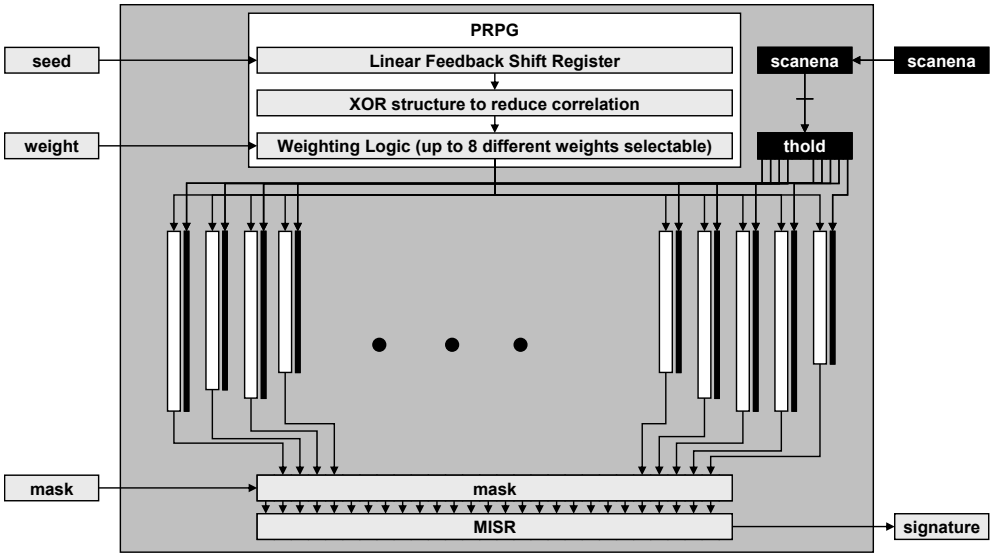

Fig. 1. STUMPS extended with scan-chain disable

and because of the diversity of the logic inside the SPE, the SPE was chosen for the case study presented below.

The SPE [23] has been designed around SIMD floating point and fixed point execution units with an operand size of 128 bits. These execution units are accompanied by a 128-entry register file and 256kB scratch pad memory called Local Store. The SPE consists of roughly 20.9M transistors, 7M in the logic parts and 14M in the memory arrays of the Local Store.

The characteristics for the SPE relevant for its test are:

- 1.8M logic gates, 7M transistors in logic, 14M transistors in arrays
- 3M faults using a stuck-at fault model in LBIST mode
- 150k latches
- 110k latches are scannable, 40k non-scannable

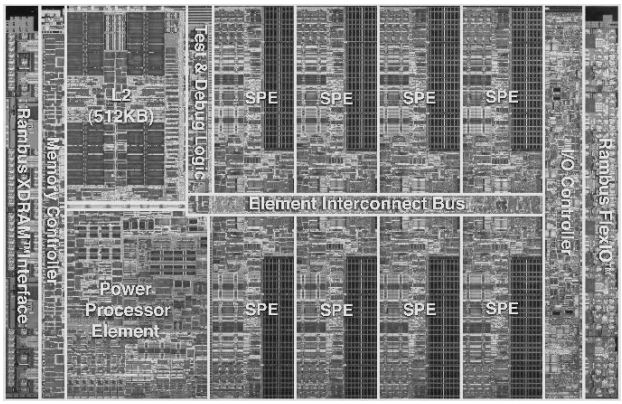

Fig. 2. Die Photo of the Cell processor

- 82.5k latches are part of 32 STUMPS channels (Self-Test Using MISR and PRPG [2])
- Memory arrays are not included in the LBIST model and instead covered by dedicated Array BIST engines

The SPE's power efficiency is mainly obtained by using extensive clock gating with the help of so called local clock buffers (LCB). In its simplest form an LCB is just a buffer that redrives the clock signal. This kind of LCB is used in regular clock tree designs, where the clock is distributed using a balanced tree of inverters. There, multiple parallel clock trees are used to distribute the different clock signals needed for level-sensitive scan design (LSSD) [24]. More advanced LCBs generate the required clock signals locally and only require a single instance clock tree (Figure 4).

Furthermore, it is not difficult to add clock gating logic to such a LCB in a controlled way, so as to avoid problems like skew or glitches. Clock gating is usually used in a multitude of ways and granularity. At the unit level, the clock signals for a whole functional unit (e.g. an execution unit) may be disabled when there is no activity in the unit. At the level of pipeline stages, only the stages containing instructions may be clocked. Even more fine grained clock gating derives the gating from the operand data itself and only activates the clocks to flip-flops that are necessary to produce the desired result [25].

B. Test characteristics of the SPE

The basic test architecture of the SPE (and the Cell processor in general) is a modification of the STUMPS scheme

proposed in [2] (See Figure 1). STUMPS is a widely used self-test methodology [26], [27] that employs multiple scan-chains in parallel. The patterns are generated by a pseudo-random pattern generator (PRPG) that consists of a linear feedback shift-register (LFSR), logic to reduce the correlation of adjacent scan-chains and a per-channel weighting logic [28] that allows for the adjustment of the distribution of logic 1s and 0s in the pattern. The pattern response is compressed by a multiple input signature register (MISR) which is preceded by a masking array that allows to mask defective scan chains or those with unknown values for diagnostic purposes.

For the various scan chains in the Cell processor, the abovementioned LCBs also generate the clocks that are used during test to scan in test patterns and consequently, clock gating can also be implemented for these clock signals. This is presently employed to do at-speed testing while scanning at a fraction of the target frequency.

The scan-clock gating is done on a scan-chain level, so that the clock can be gated on a chain-by-chain basis. The state of the scan-gating is stored in a central register (the scan-enable register $scanena$, Fig 1) that is located in a configuration scan-ring together with the seed and weights of the PRPG and the signature of the MISR. Therefore, it can easily be set while reading the MISR signature or setting seed and weights. Each bit in the scan-enable register corresponds to one scan-chain of the STUMPS and is distributed to the LCBs' $testhold\_b$ input in Figure 4.

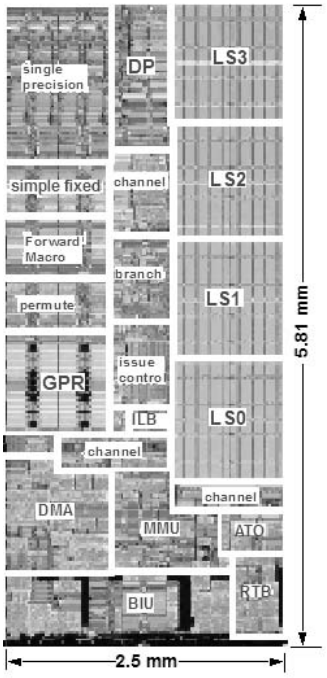

Fig. 3. SPE Synergistic Processing Element of the Cell processor

III. Low power test planning

The test planning algorithm described below determines sets of scan chains to be activated during certain time steps. The granularity of these time steps has impact on both amount of test data and power saving. If the granularity was on a pattern by pattern basis, the enabling information would have to be provided separately for each pattern and we would loose the BIST capability. Hence, the number N of patterns for which the set of activated scan chains is not altered, will be larger than 1, and the experiments discussed below are performed by using $N = 1024$ patterns for one configuration (also called test block).

The appropriate test plan is derived in three steps:

1) Determining a set T of essential patterns.
Without any scan chain disabling, fault simulation is performed with all the patterns generated by the STUMPS configuration from a given LFSR seed initialization. Only those patterns are included in T, which detect at least one new fault previously not detected. T will be a very small subset of all the patterns generated by the STUMPS architecture, and it could be further reduced by reverse or even permuted fault simulation. The smaller T will be, the better the expected power savings will be, too.

However, reverse or permuted fault simulation increase computing time significantly and are not supported by all commercial fault simulators so that extensive scripting would

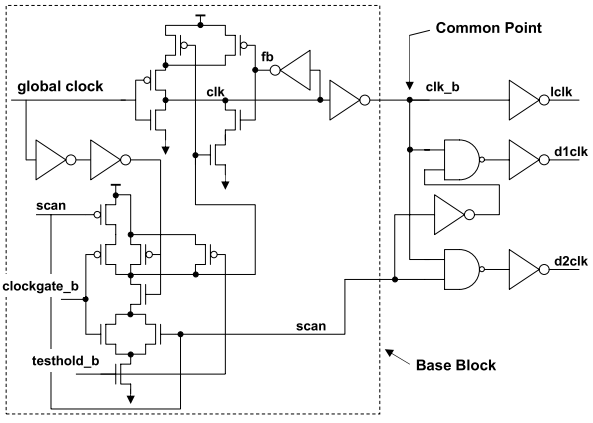

Fig. 4. LCB Design in the Cell Processor [22]

be required. Therefore, all results reported below are obtained by single pass fault simulation.

2) Determining test units.

A test unit consists of a test pattern and a set of flip-flops, which have to be controlled and observed when this pattern is applied. For each pattern $t \in T$ let F_t be the set of faults, which are detected by t the first time. Each fault $f \in F_t$ determines an input cone which includes all the flip-flops, which must be set in order to control the fault site. A second set of flip-flops is determined be the output cone of the fault site, these are all flip-flops reachable from f where the fault could be observed.

In order to propagate the fault signal to one of the output flip-flops, additional input flip-flops must be controlled. Hence, the essential flip-flops $e(f)$ of fault f are all the flip-flops of the output cone plus all input flip-flops which are initial point of a path to at least one output flip-flop (See Figure 5). (This combination of input and output cones is also sometimes called support region [29] of f.) The essential flip-flops are in this case $e(f) = \{FF_2..FF_7, FF_{11}..FF_{14}\}$, and for testing f the scan chains $SC1$ and $SC2$ have to be activated.

A test unit $U_t = \{t, FF_t\}$ consist of a test pattern t, and the union of all the essential flip-flops of the faults that are detected for the first time by t:

$$FF_t = \bigcup_{f \in F_t} e(f)$$

All the scan chains, which contain at least one flip-flop of FF_t will have to be enabled, if the pattern t is applied. Hence, the test unit U_t determines just one pattern and the set of required scan chains. In the case of Figure 5, these chains are $SC1$ and $SC2$.

3) Determining test blocks.

As already pointed out, a scan configuration has to be applied for a sequence of patterns $< t_1, ..., t_N >$.

A test block is just such a pattern sequence, and the union of all the flip-flops determined by the corresponding test units:

$$B = (< t_1, ..., t_N >, \bigcup_{i=1..N} FF_{t_i})$$

Again, the essential flip-flops of a block

$$FF_B = \bigcup_{i=1..N} FF_{t_i}$$

determine the scan chains to be enabled when applying the N patterns starting from the given LFSR initialization. The encoded information for this scan chain configuration will be shifted in together with the LFSR seed initiating the sequence of effective patterns $< t_1...t_N >$.

After repeating the steps above for each Block of 1024 patterns, the outcome of the procedure described above is a test plan which consists of a set of test blocks $\{B\}$, each block corresponds to a seed, a scan configuration and finally a certain amount of average power which can be computed with various commercial tools.

IV. Results, goals, and limits

In this section, we evaluate the method presented above with respect to the goals listed in the introduction.

A. Power savings

As mentioned before, most existing BIST power reduction techniques exhibit an unacceptable computational complexity. Besides the generally high complexity of some of the used algorithms, most of the techniques use, in one way or another, some kind of gain function that is based on power simulation of the intermediate solutions. Very often, the weighted witching activity of every net is computed during simulation in order to estimate the power during test.

Unfortunately this method does not scale very well because it conflicts with the optimization that is done by commercial test simulators. The test simulator does not simulate the scan process clock for clock, instead it skips the tedious scanning of

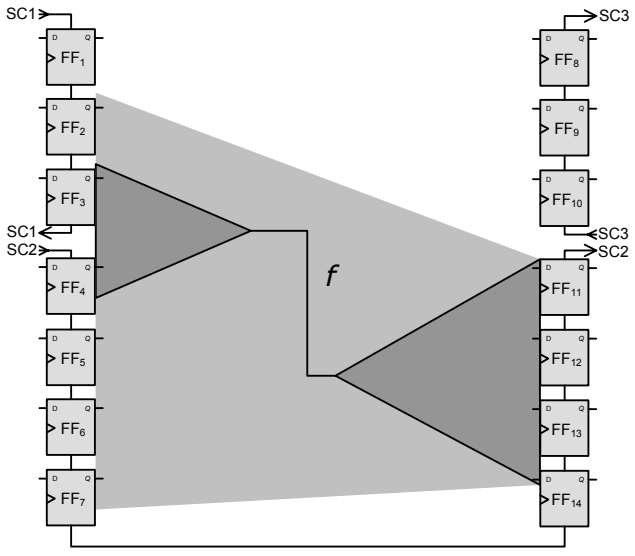

Fig. 5. Test units for fault f

the patterns and directly forces the pattern values into the flip-flops in the STUMPS channels. However, the purpose of this work is to especially reduce the power used during *scanning*, which makes up for about 99% of the power consumption during test. So in order to use this power simulation approach to simulate the power consumption during test, the circuit simulator had to perform a simulation that includes the scanning of patterns. Hence, this simulation would take about 1000 times as many simulation cycles when compared to a regular logic simulation.

On the other hand, at least in large circuits, it can be realistically assumed, that the power consumption of the logic that is driven by a certain scan chain is proportional to its length in terms of flip-flops. Using this assumption we can estimate the power consumption using the number of flip-flops that receives a clock signal (i.e. flip-flops in enabled scan-chains) during a certain time period. This number is in relation to the power consumption when all flip-flops receive a clock signal (maximum average power) and when no scan flip-flops are clocked (just static power).

Figure 6 shows the estimated reduction in flip-flops clock cycles (i.e. the number of clock events summarized over all flip-flops). The proposed approach reduces the dynamic energy consumption of the whole test by 39.0%.

Initial point for this result was a set of 200k patterns and the planning algorithm computing the scan configuration for each block of $N = 1k$ patterns. Increasing the number of patterns may not be beneficial in a manufacturing environment but would provide even better results of this method.

Computation time is mainly dominated by the time required for the fault simulation of the circuit, which is in the order of days. In contrast, the CPU time required for computing the set of activated scan-chains for all blocks is only a few hours.

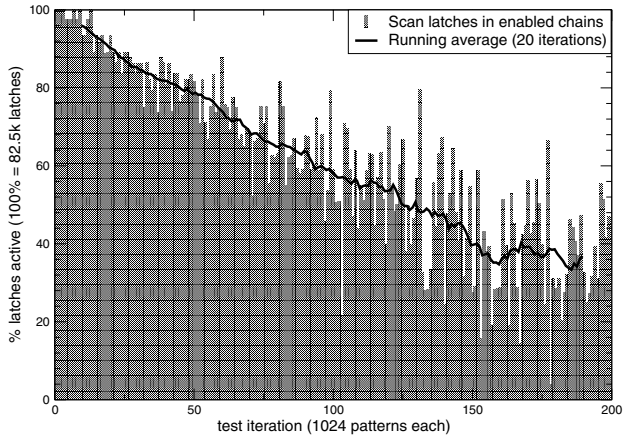

Fig. 6. Power consumption

B. Test length and fault coverage

The method presented here uses the original test set and does not add any patterns but only the scan enabling information for each block. Since this information may be scanned in in parallel with the evaluation of the signature, extra cycles are not required. Test time may be even shortened as due to the reduced power consumption a higher frequency may be an option.

The method does not change the coverage of stuck-at faults at all. However, since now scan chains are disabled during test one may be concerned about the detection of non-target faults. The same situation arises for scan chain masking to prevent unknown values in the signature register. It has been shown that the detection of non-target faults can be improved by allowing multiple observation points for one stuck-at fault, either multiple test patterns for one fault or multiple outputs, and the detection rate saturates rather fast already for a low multiplicity of 2 or 3 [30]. This is one of the reasons why in section III test units are determined by the complete output cone of a fault and not just by a single flip-flop where the stuck-at fault can be observed. Moreover, the work presented here may also be applied to more complex test sets without changes, e.g. delay tests.

C. Flexible test block scheduling

Each test block is specified by a seed of the LFSR and the scan enabling information, both provided by the external tester. Hence, there is complete freedom to reorder the blocks. The order does not have impact on the overall energy, but has impact on the heat profile. While a single test block takes just a few milliseconds and does not contribute much to heating (even for test blocks with very high power dissipation), several test blocks with high acitvity may have impact and may enforce a reduction of the test speed. In order to avoid this, a test schedule is preferred which reduces the average power also in smaller time windows, for example limit the average power for each k consecutive blocks (see Figure 7).

While scheduling for a balanced power consumption allows a uniform test speed along the entire test course, a more aggressive scheduling is possible, too. It may be advantageous to schedule the more power hungy blocks at the beginning of the test procedure in order to exploit the fact that the die is still cold (Figure 8). Moreover, the reordering divides the complete

test into different phases so that scanning frequency may be increased toward the end of the test process.

D. Design flow

The design flow used for large microprocessor designs has a high complexity and usually requires heavy interaction of many different software tools, both proprietary closed source software and home-grown tools. Integration into such a design flow and all its peculiarities means interfacing with all these different tools. Moreover, dealing with software that does not provide easy access to its internal programming interfaces might also prove a limitation in that functionality might have to be replicated.

Innovating in a production environment also means to provide several fall back paths in case of design bugs. In this special case this means to leave the existing test methodology as unaltered as possible, thus providing a fall-back to the regular LBIST. This also assures, that investments into existing methodologies, tools and know-how can be reused as far as possible. Other considerations when implementing a new test technique are its robustness against calculation or design errors.

Since the method presented does not require any design changes on top of the DfT logic already implemented in the Cell processor, it fits perfectly into such a production environment. In order to enhance design productivity and compliance with CAD tools several compromises had to be taken, which limit the power savings obtained:

- The fault simulation tool did not deliver the set of outputs where a fault was observable. Hence, the complete output cone had to be considered when creating the test units, which in turn increased the number of enabled scan chains.
- For each pattern and fault, it was not evaluated which bits are essential and which bits may be left unspecified. There are methods for pattern stripping proposed in literature for determining a minimal set of specified bits [31]. If only scan chains are enabled which contain specified bits instead of the complete input cones, further saving will be obtained. Unfortunately, commercial tools do not provide this information, and the algorithms proposed so far do not handle multi-million gate circuits.

V. Conclusion

For the Cell processor, a test planning and test scheduling method has been developed, which decreases average test power significantly by enabling in each test phase only a subset of the scan chains. The method has no or neglectable impact regarding the number of test patterns, the test setup time, the amount of hardware overhead and the test equipment that is required, while significantly reducing the energy consumed during Logic BIST.

VI. Acknowldegements

The authors gratefully acknowledge the many contributions from Michael Kessler and Steven Ferguson at IBM as well as the entire Cell processor team at IBM Boeblingen. Also, we would like to thank Tobias Bergmann at the Institut fuer Technische Informatik for valuable discussions regarding low-power design.

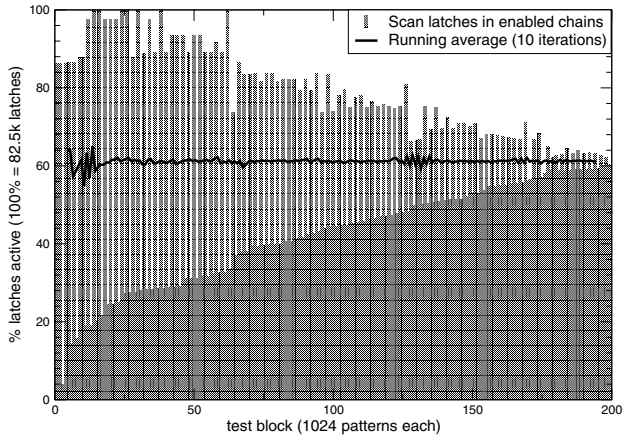

Fig. 7. Balanced power consumption

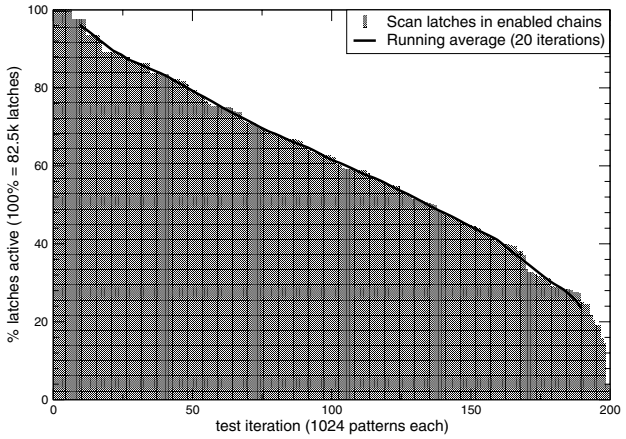

Fig. 8. Enabled scan flip-flops using reordering

REFERENCES

[1] T. W. Williams and E. B. Eichelberger, "Random patterns within a structured sequential logic design," in *Digest of Semiconductor Test Symposium, Cherry Hill, NJ, October 25-27, 1977*, 1977, pp. 19–28.

[2] P. H. Bardell and W. H. McAnney, "Self-testing of multichip logic modules," in *Proceedings International Test Conference 1982, Philadelphia, PA, USA, November 1982*, 1982, pp. 200–204.

[3] Y. Zorian, "A distributed BIST control scheme for complex VLSI devices," in *Proceedings of the 11th IEEE VLSI Test Symposium (VTS)*, 1993, pp. 4–9.

[4] J. Saxena, K. M. Butler, V. B. Jayaram, S. Kundu, N. V. Arvind, P. Sreeprakash, and M. Hachinger, "A case study of ir-drop in structured at-speed testing," in *Proceedings 2003 International Test Conference (ITC 2003), Breaking Test Interface Bottlenecks, 28 September - 3 October 2003, Charlotte, NC, USA*, 2003, pp. 1098–1104.

[5] S. Gerstendoerfer and H.-J. Wunderlich, "Minimized power consumption for scan-based BIST," in *Proceedings IEEE International Test Conference 1999, Atlantic City, NJ, USA, 27-30 September 1999*, 1999, pp. 77–84.

[6] S. Ghosh, E. MacDonald, S. Basu, and N. A. Touba, "Low-power weighted pseudo-random BIST using special scan cells," in *Proceedings of the 14th ACM Great Lakes Symposium on VLSI 2004, Boston, MA, USA, April 26-28, 2004*, 2004, pp. 86–91.

[7] N. Parimi and X. Sun, "Toggle-masking for test-per-scan VLSI circuits," in *19th IEEE International Symposium on Defect and Fault-Tolerance in VLSI Systems (DFT 2004), 10-13 October 2004, Cannes, France, Proceedings*, 2004, pp. 332–338.

[8] P. Girard, L. Guiller, C. Landrault, and S. Pravossoudovitch, "A test vector inhibiting technique for low energy BIST design," in *17th IEEE VLSI Test Symposium (VTS '99), 25-30 April 1999, San Diego, CA, USA*, 1999, pp. 407–412.

[9] S. Manich, A. Gabarro, M. Lopez, and J. Figueras, "Low power BIST by filtering non-detecting vectors," *Journal of Electronic Testing Theory and Applications (JETTA)*, vol. 16, no. 3, pp. 193–202, 2000.

[10] L. Whetsel, "Adapting scan architectures for low power operation," in *Proceedings IEEE International Test Conference 2000, Atlantic City, NJ, USA*, 2000, pp. 863–872.

[11] S. Bhunia, H. Mahmoodi-Meimand, D. Ghosh, and K. Roy, "Power reduction in test-per-scan BIST with supply gating and efficient scan partitioning," in *6th International Symposium on Quality of Electronic Design (ISQED 2005), 21-23 March 2005, San Jose, CA, USA*, 2005, pp. 453–458.

[12] S. Chakravarty and V. Dabholkar, "Minimizing power dissipation in scan circuits during test application," in *Proceedings IEEE International Workshop on Low Power Design 1994, Los Alamitos, CA, USA*, 1994, pp. 51–56.

[13] V. Dabholkar, S. Chakravarty, I. Pomeranz, and S. Reddy, "Techniques for minimizing power dissipation in scan and combinational circuits during test application," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 17, no. 12, pp. 1325–1333, 1998.

[14] Y. Bonhomme, P. Girard, L. Guiller, C. Landrault, S. Pravossoudovitch, and A. Virazel, "Design of routing-constrained low power scan chains," in *2004 Design, Automation and Test in Europe Conference and Exposition (DATE 2004), 16-20 February 2004, Paris, France*, 2004, pp. 62–67.

[15] P. Girard, L. Guiller, C. Landrault, and S. Pravossoudovitch, "Low power BIST design by hypergraph partitioning: Methodology and architectures," in *Proceedings IEEE International Test Conference 2000, Atlantic City, NJ, USA, October 2000*, 2000, pp. 652–661.

[16] S. Wang and S. K. Gupta, "DS-LFSR: A new BIST TPG for low heat dissipation," in *Proceedings IEEE International Test Conference 1997, Washington, DC, USA, November 3-5, 1997*, 1997, pp. 848–857.

[17] S. Wang and S. K. Gupta, "LT-RTPG: a new test-per-scan BIST TPG for low heat dissipation," in *Proceedings IEEE International Test Conference 1999, Atlantic City, NJ, USA, 27-30 September 1999*, 1999, pp. 85–94.

[18] F. Corno, M. Rebaudengo, M. S. Reorda, G. Squillero, and M. Violante, "Low power BIST via non-linear hybrid cellular automata," in *18th IEEE VLSI Test Symposium (VTS 2000), 30 April - 4 May 2000, Montreal, Canada*, 2000, pp. 29–34.

[19] R. Sankaralingam, N. A. Touba, and B. Pouya, "Reducing power dissipation during test using scan chain disable," in *19th IEEE VLSI Test Symposium (VTS 2001), 29 April - 3 May 2001, Marina Del Rey, CA, USA*, 2001, pp. 319–325.

[20] P. Girard, L. Guiller, C. Landrault, S. Pravossoudovitch, and H.-J. Wunderlich, "A modified clock scheme for a low power BIST test pattern generator," in *19th IEEE VLSI Test Symposium (VTS 2001), 29 April - 3 May 2001, Marina Del Rey, CA, USA*, 2001, pp. 306–311.

[21] M. Riley, L. Bushard, N. Chelstrom, N. Kiryu, and S. Ferguson, "Testability features of the first-generation Cell processor," in *Proceedings of the IEEE International Test Conference (ITC), 8-10 Nov. 2005, Austin TX*, 2005, p. 6.1.

[22] D. Pham, S. Asano, M. Bolliger, M. N. Day, H. P. Hofstee, C. Johns, J. Kahle, A. Kameyama, J. Keaty, Y. Masubuchi, *et al.*, "The design and implementation of a first-generation Cell processor," in *International Solid-State Circuits Conference (ISSCC) Digest of Technical Papers, 6-10 Feb. 2005, San Francisco CA*, 2005, pp. 184–185.

[23] B. Flachs, S. Asano, S. Dhong, P. Hofstee, G. Gervais, R. Kim, T. Le, P. Liu, J. Leenstra, J. Liberty, *et al.*, "A streaming processing unit for a Cell processor," in *International Solid-State Circuits Conference (ISSCC) Digest of Technical Papers, 6-10 Feb. 2005, San Francisco CA*, 2005, pp. 134–135.

[24] E. B. Eichelberger and T. W. Williams, "A logic design structure for LSI testability," *Journal of Design Automation Fault-Tolerant Computing*, vol. 2, no. 2, pp. 165–178, 1978.

[25] H. M. Jacobson, P. Bose, Z. Hu, A. Buyuktosunoglu, V. V. Zyuban, R. Eickemeyer, L. Eisen, J. Griswell, D. Logan, B. Sinharoy, and J. M. Tendler, "Stretching the limits of clock-gating efficiency in server-class processors," in *11th International Conference on High-Performance Computer Architecture (HPCA-11 2005), 12-16 February 2005, San Francisco, CA, USA*, 2005, pp. 238–242.

[26] B. L. Keller and T. J. Snethen, "Built-in self-test support in the IBM engineering design system," *IBM Journal of Research and Development*, vol. 34, no. 2/3, pp. 406–415, 1990.

[27] G. Hetherington, T. Fryars, N. Tamarapalli, M. Kassab, A. S. M. Hassan, and J. Rajski, "Logic BIST for large industrial designs: real issues and case studies," in *Proceedings IEEE International Test Conference 1999, Atlantic City, NJ, USA, 27-30 September 1999*, 1999, pp. 358–367.

[28] H.-J. Wunderlich, "Multiple distributions for biased random test patterns," in *Proceedings International Test Conference 1988, Washington, D.C., USA, September 1988*, 1988, pp. 236–244.

[29] I. Hamzaoglu and J. H. Patel, "New techniques for deterministic test pattern generation," in *16th IEEE VLSI Test Symposium (VTS '98), 28 April - 1 May 1998, Princeton, NJ, USA*, 1998, pp. 446–452.

[30] Y. Tang, H.-J. Wunderlich, H. Vranken, F. Hapke, M. Wittke, P. Engelke, I. Polian, and B. Becker, "X-masking during logic bist and its impact on defect coverage," in *Proceedings IEEE International Test Conference (ITC 2004), October 25-28, 2004, Charlotte, NC, USA*, 2004, pp. 442–451.

[31] S. Kajihara and K. Miyase, "On identifying don't care inputs of test patterns for combinational circuits," in *International Conference on Computer-Aided Design ICCAD, November 4-8, 2001, San Jose, CA, USA*, 2001, pp. 364–369.