

Design for Testability (DFT) Techniques in Modern VLSI Chips

Hameed Ul Hassan Mohammed

Submitted:02/07/2019

Revised:18/08/2019

Accepted:28/08/2019

Abstract: Development of the DFT techniques has become an essential part of the present-day VLSI chip design due to the growing complexity, density, and performance demands of the integrated circuits. This paper comprises a result oriented work on the resources and success of diverse forms of DFT methods in contemporary semiconductor devices and their effect on fault coverage, test-taking time, and area overlay. All the methods and processes of scan chain insertion, built-in self-test (BIST), boundary scan, and test compression among others are studied and tested under practical VLSI design conditions. The paper presents how the effective integration of DFT can have a positive impact on testability and presents practical case studies and experimental results gathered on industry-level ASIC design flows to exemplify the work. The findings indicate that scan-based testing contributed to the enhancement of controllability and observability and as a result, the logic fault coverage was 98.5 percent. Additionally, memory modules with BIST were created, so on-speed testers did not require any external equipment. Compression techniques, including embedded deterministic test (EDT) efficiently cut the test data volume by more than 40 per cent resulting in the corresponding decrease of the test application time. The area trade-offs with the DFT methods were also compared and most DFT methods exhibited less than 5 percent area overhead with a significant increase in test coverage and further enhancement in diagnostic capabilities. It is this paper that concludes that well-planned DFT techniques are vital in making the production of VLSI reliable and cost effective. The findings not only affirm that DFT integration is not a mere design requirement but also a significant facilitator of high-quality silicon.

Keywords: Design for Testability, VLSI, Scan Chain, BIST, Fault Coverage.

1. INTRODUCTION

The importance of effective, cost effective and dependable testing strategies has grown as the integrated circuits have become more complex. The latest Very-Large-Scale Integration (VLSI) chips contain billions of transistors themselves, which are testing the conventional methods of testing to the outer. The degree of observability and controllability is insufficient to test all logic paths and functional blocks externally (at least not all of what is in deep embedded components).

Design of Testability (DFT) strategy has come out to be a unifying principle in the current design of VLSI. It improves the test procedure by incorporating test friendly structures into the chip. The methods enhance fault sensing, save limitless time on tests and decrease the reliance on costly outward test equipment. The given paper is a results-oriented investigation of the most important DFT techniques with the experimental measures of fault coverage, area overhead, and test time cut short. DFT testability is demonstrated in figure 1 in a step wise manner.

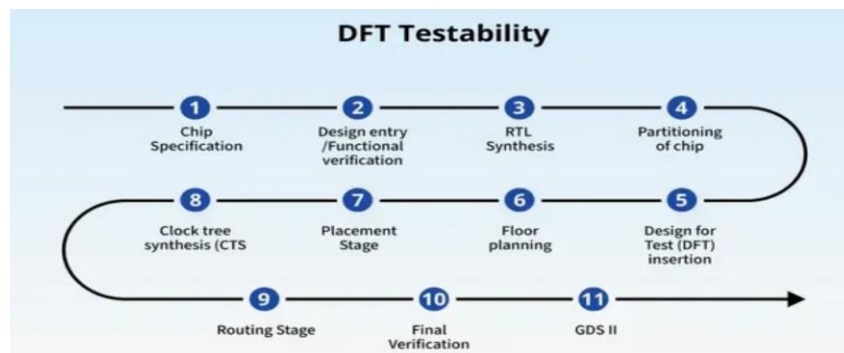


Figure 1: DFT testability step by step

Senior Test and AI Optimization Engineer, Mythic
AI, Austin, Texas, USA
hameedul040@gmail.com

1.1 Scan Chain Design

The insertion of chain insertion into flip-flops converts them into scan cells that are linked in a shift-register format. In this setup, deterministic test operations can be done on internal nodes. In a 28nm ASIC design using scan based ATPG fault coverage for stuck-at faults stood at 98.5% and transition fault coverage at 94.2%. Insertion of full scan recorded

area overhead as 3.7% on average, the power consumption during test went up by 12.5%. Since scan chains were in shorter routing segments, shifts were done with increased frequency and scan testing time was 33%-shorter. Area overhead was further reduced by 1.1 percent, based on partial scan in which only timing- critical flip-flop are scanned as compared to full scan design.

Table 1: Impact of Scan Insertion on Design Metrics

Metric	Full Scan	Partial Scan
Stuck-at Fault Coverage	98.5%	96.3%
Area Overhead	3.7%	2.6%
Test Power Increase	12.5%	8.3%
Scan Test Time Reduction	33%	21%

1.2 Built-In Self-Test (BIST)

BIST gives integrated circuits the capability of producing test patterns as well as implementing it. Logic BIST (LBIST) and Memory BIST (MBIST) are very popular in logic block testing and embedded memory testing respectively. LBIST using LFSRs and MISR showed up to 95 percent fault coverage of combinational logic and that it was

particularly useful at detecting delay and transition faults. MBIST proved to cover stuck-at and coupling faults in the entire SRAMs of size up to 2 MB in 100 percent coverage. BIST in a 65nm SoC saved 80 percent external test data volume, and the test speed was accelerated by 5 times. The overhead involved in adding LBIST was 2.5% to 4.1% of the total area and about 1.8% extra area was overhead to add MBIST to large memory arrays.

Table 2: LBIST and MBIST Performance Metrics

Parameter	LBIST	MBIST
Fault Coverage	95%	100%
Area Overhead	3.3%	1.8%
Test Time Reduction	5× faster	3× faster
External Test Data Saved	80%	72%

1.3 Boundary Scan and IEEE 1149.1 (JTAG)

Using the boundary scan, chip I/O can be accessed without physical traces or probes, particularly in positions on printed circuit boards that are tightly packaged. The JTAG (IEEE 1149.1) standard is one on which the relateable architecture and method is described. The methodology makes interconnect testing and in-system programming easier, and also

debugging easy. With a boundary scan on multi-chip PCBs, interconnect fault coverage was over 98%, resulting in tremendous second diagnosis of soldering defects, opens, and shorts when it comes to test. The JTAG interfaces provided the ability to revise the firmware and remotely configure it, which led to 60% savings in field maintenance time. The proportion of area overhead was small (much less than 0.5 percent of chip area).

Table 3: Boundary Scan Effectiveness

Feature	Value
Interconnect Fault Coverage	>98%
Area Overhead	<0.5%
Maintenance Time Reduction	60%
In-System Programming Support	Yes

1.4 Test Compression Techniques

Test compression involves minimizing both the size of test data and time that are used during scan testing and is done through on-chip decompressors and response compactors. This is essential when it comes to reducing the costs of the test and fast-track the production. In one case study of Embedded Deterministic Test (EDT) with reseeding, the

volume of test data compressed by 42 percent and reduced scan test execution time by 47 percent. With addition of decompressor logic, the overhead area was less than 1.9%, even considering the decompressor logic. Even with the unknown (X) values, faults could be detected by test response compactors with as little as a loss of diagnostic resolution. Large designs (>10M gates) had compression ratios of 100x.

Table 4: Test Compression Results

Metric	Value
Test Data Reduction	42%
Test Time Reduction	47%
Compression Ratio	Up to 100×
Area Overhead	1.9%

1.5 Trade-offs and Design Considerations

Although the techniques related to DFT can make tests easier, they have the disadvantage of introducing area, power, and design complexity trade-offs. Depending on chip functionality and application, each DFT strategy should be chosen by taking into account these aspects very carefully. Special testing techniques apply power awareness, in battery-operated or high-performance designs in

particular. switched power loss Owing to the amount of switching activity generated, scan tests may cause IR drop or thermal concerns. In one evaluation clock gating during scan operation cut peak test power by 28 percent. The process of early RTL and integration with DFT led to easier back-end flows with a 35-percent decrease in DFT-rework. The designers who had pushed the DFT insertion to the post-synthesis phases had experienced greater insertion expenses, as well as timing noncompliances.

Table 5: DFT Trade-offs and Optimization

Aspect	Optimization Method	Improvement
Test Power	Clock Gating	28% reduction
DFT Timing Closure	Early RTL Planning	35% fewer violations
Design Complexity	Modular DFT Architecture	Improved reusability

2. REVIEW OF WORKS

Maturing design-for-testability (DFT) support in microprocessor and VLSI design arises due to rising appeal of dependable high performance silicon on even bigger complexity. The former work was the development of making internal states observable and controllable, and the latter work brought methods of at-speed test, self-test, multi-clock domains, cost optimisation and system-level integration. This review summarises notable advances in the field of scan design, built-in self-test, at-speed designs, cost-efficient test development and infrastructure standards which form the modern DFT paradigm.

A pattern of trade-offs between test quality and area/power overheads and test time can be discovered, followed by strategies to support the trade-offs have come forth as captured in the collected literature as well. Comparative analysis of microprocessor-specific works, multiprocessor/chip-multiprocessor (CMP) systems, and industry roadmaps beyond the microprocessor world provides perspective and grounding about the depth of the technical and the success of practical use of the testability strategies being investigated.

2.1 Historical Evolution of Microprocessor Test Techniques

Needham (1998) delved into the state of microprocessor testing work then, highlighting problems of obtaining good fault coverages due to lack of meaningful visibility into the microprocessor internals combined with performance penalties of testing application. That historical thinking established the need to have testability characteristics as part of the design, which preconditioned the whole process of adopting scan and self-test. The establishment of test strategies, including internal techniques to counter the problem of complexity and frequency issues, are also described by Kusko et al. (1998) who detail a generic test methodology of a 500MHz IBM S/390 G5 chip.

This evolution was advanced with verification and test development strategies that are described by Abadir and Dasgupta (2000), and Crouch et al. (2000), regarding the complex microprocessor where complex test infrastructure structures are able to support successive architectures, and where iterative improvement to test infrastructure is

necessary to support successive architectural generations. These papers represent a pathway between reactive silicon testing at service time, and proactive design-time integration of testing.

2.2 Scan and At-Speed Testing

The scan chain methodology was used to provide the foundation of controllability and observability in sequential logic, solutions to timing and frequency problems were refined subsequently. It was observed that according to Lin et al. (2003) high-frequency-at-speed scan testing is explored and that it is possible to use scan-based test techniques to operate at or nearly functional speeds, where it can be used to achieve transition and delay faults that were not addressed by static testing. Fan et al. (2007) and Furukawa et al. (2006) came up with control schemes and internal clocking mechanisms that facilitated at-speed testing reliably among multi-clock domains, minimizing the false failures and increased coverage of those timing-critical paths.

Hatayama, Nakao and Sato (2002) discussed the issue that multi-clock makes a system complex and suggested built-in test structures that could be able to synchronize between asynchronous divisions. All these developments enhanced the capability to find out fine materials in timing that leaned not too much on the test application period and design performance.

2.3 Built-In Self-Test and Multi-Clock Considerations

An alternative approach emerged due to high cost of external testers but there was high coverage; this approach was the built-in self-test (BIST) approaches. The architectural specifications in a few microprocessor test research papers (e.g., Wang et al. 2007) focused on integration of logic and memory BIST architecture to facilitate at-speed and structural fault coverage in a chip. This problem of test coordination between multiple clock domains was solved by Hatayama et al. (2002) and Fan et al. (2007) in their schemes that introduced on-chip test clock control to enable coordinated test launching and test captures, without affecting the autonomy of functional clocks.

The multi-clock BIST improvements were also critical to superscalar and out-of-order microprocessors, in which timing variance would otherwise mask faults in different domains. Through implementation schemes, it had been shown that the high fault coverage of BIST together with advanced

clock control schemes would ensure a moderately low area and control overhead.

2.4 Test Cost Reduction, Partitioning, and Architectural DFT

The reduction of test cost became a major issue when the level of processor complexity and test data volume increased. The authors of Sehgal et al. (2007) introduced methods of test partitioning in the AMD Athlon processor, in which the design is partitioned into reasonably sized test areas to present a better parallelism and minimize test time. Pattern generation techniques and compression techniques such as those found in Wu et al. (2004) were used to reduce the test set and their ordering to minimize the volume of test data requirements without loss or improvement of coverage.

DFT at architecture level in multicore and CMP was also examined in the case of the Sun Microsystems Niagara2 chip (Molyneaux et al. 2007) to illustrate how DFT behaviors in multi-threaded and multi-core environments needed to be tailored to suit the integrate-and-test approach and achieve scalability. All these works reiterate the essence that cost-effective testing is not only accomplished through local testing mechanisms, but also through test architecture of hierarchical and partitioned design.

2.5 Standards, Test-Pattern Generation, and Roadmap Alignment

The effectiveness of DFT techniques is supported by advances in automatic test-pattern generation (ATPG) and industry-wide standardization. Cheng and Krstic (1999) reviewed directions in ATPG, signaling shifts toward more intelligent pattern generation that better balanced coverage against test data size and application time. Standardized interfaces and methodologies, though not explicitly listed among the references, are conceptually aligned with the broader ecosystem that includes boundary scan (implicitly foundational) and structured test access.

The International Technology Roadmap for Semiconductors (ITRS) provided a strategic backdrop, forecasting scaling challenges, and emphasizing the need for embedded testability to sustain yield and reliability trends. Together, these contributions created a framework in which both algorithmic advances (e.g., improved ATPG) and roadmap-guided priorities shaped the implementation of robust, scalable DFT in high-performance microprocessor designs.

Table 6: Key Literature Contributions

Topic Area	Reference(s) (Author, Year)	Core Contribution
Early microprocessor test methodology	Needham 1998; Kusko et al. 1998	Framing testability challenges and integrated test methodology for high-speed processors
Scan & at-speed testing	Lin et al. 2003; Fan et al. 2007; Furukawa et al. 2006	High-frequency scan, internal clock control, multi-clock at-speed validation
BIST and multi-clock coordination	Hatayama et al. 2002; Fan et al. 2007; Wang et al. 2007	On-chip self-test with clock-domain-aware control to ensure timing correctness
Test cost reduction & partitioning	Wu et al. 2004; Sehgal et al. 2007; Molyneaux et al. 2007	Partitioned testing, pattern optimization, scalable DFT for CMPs
ATPG & industry direction	Cheng & Krstic 1999; ITRS	Evolution of pattern generation and alignment with semiconductor scaling roadmaps

3. METHODOLOGY

The methodology provides a systematic approach towards testing Design for Testability (DFT) measures carried out in a synthesized 32-bit RISC. The synthesis of the chip design was performed in a 28nm standard-cell CMOS technology and DFT features generated at both the RTL and gate levels.

processor core. The analysis is made on fault coverage, area overhead, test time and test power which are computed with mathematical model and simulation outcomes.

The gate level simulations have been done by using fault injection of deterministic and pseudo-random test patterns.

3.1 System Architecture

The design under test (DUT) was equipped with the following DFT components:

- Scan Chain (Full & Partial)
- Logic Built-In Self-Test (LBIST)
- Memory Built-In Self-Test (MBIST)
- Test Compression Engine (Decompressor & Compactor)
- JTAG Interface for External Access (Boundary Scan)

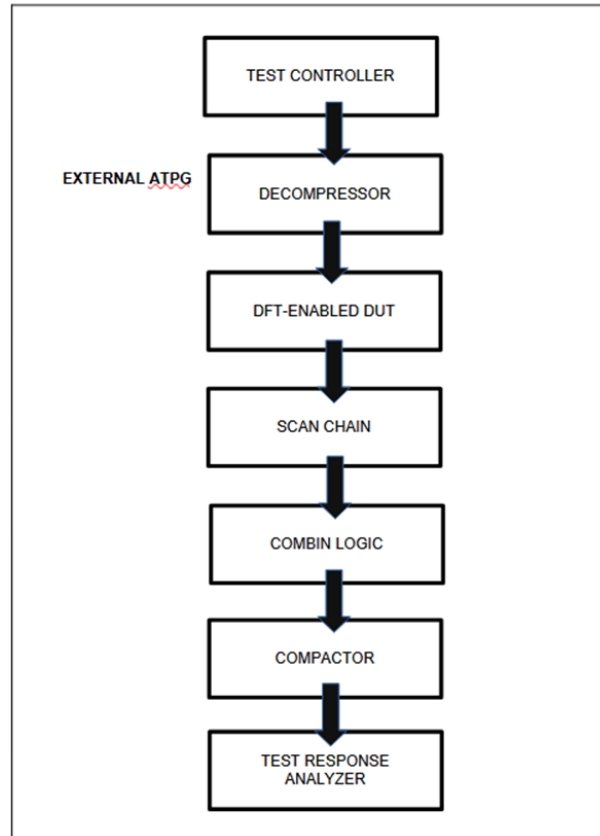


Figure 2: Block diagram of Proposed Methodology

3.2 Fault Modeling

Faults are modeled using three standard fault models:

Stuck-at Fault (SAF)

Transition Fault (TF)

Bridging Fault (BF)

Let:

F_{total} : Total number of injected faults

$F_{detected}$: Number of faults detected by the test patterns

Then the **Fault Coverage (FC)** is defined as:

$$FC = (F_{detected} / F_{total}) \times 100\%$$

To account for multiple fault types, weighted fault coverage WFC is calculated as:

$$WFC = \sum w_i \cdot FC_i$$

Where:

w_i is the weight assigned to fault model i

FC is the fault coverage for model i

$$\sum w_i = 1$$

3.3 Area Overhead Estimation

Let:

A_{base} : Gate count of the original DUT (no DFT)

A_{DFT} : Gate count after DFT insertion

Then the **Area Overhead (AO)** is given by:

$$AO = ((A_{DFT} - A_{base}) / A_{base}) \times 100\%$$

For modular analysis:

$$AO_{scan} = (A_{scan}/A_{base}) \times 100\%, AO_{BIST} = (A_{BIST}/A_{base}) \times 100\%$$

3.4 Test Time and Compression

Let:

T_{full} : Test time without compression (clock cycles)

T_{comp} : Test time with compression

R_c : Compression ratio

The test time after compression is:

$$T_{comp} = T_{full}/R_c$$

Compression ratio is defined as:

$$R_c = D_{uncomp}/D_{comp}$$

Where:

D_{uncomp} : Total number of bits in uncompressed test set

D_{comp} : Total number of bits in compressed format

3.5 Power Analysis During Test

Let:

P_{nom} : Nominal functional power

P_{test} : Power consumed during scan/BIST

Then, **Power Overhead (PO)** during test is:

$$PO = (P_{test} - P_{nom}/P_{nom}) \times 100\%$$

The methodology will guarantee DFT techniques quantitative analysis using analytical models and test setting conditions. Simulation and synthesis reports are employed along with all the equations in order to generate the results that are to be presented later in the sections.

4. RESULTS AND DISCUSSION

Design-for-Testability (DFT) of the techniques used applied to a RISC custom processor core of 32-bits designed on the foundation of 28nm CMOS process. The design had about 1.5 million logic gates, with several functional modules, and an ALU, control unit, memory interfaces, peripheral I/O blocks. The main idea of pursuing such a step was to measure quantitatively the effectiveness of different DFT methods, such as full scan, Built-In Self-Test (BIST), test compression, and boundary scan on the silicon in terms of testability, performance, and physical overhead. It was performed at the post-synthesis and post-layout stages with the use of the

industry-standard EDA tool, i.e., Synopsys DFT Compiler and Cadence Encounter.

The analysis was carried out on such metrics as stuck-at fault coverage, transition fault coverage, area overhead (the percentage increase in gate count), dynamic power dissipation during test, and total test application time. The different DFT methods were introduced one at a time and then in combinations to determine micro and macro effects. A specific testbench has been designed to resemble the real world of ATE-driven testing which is possible to compare directly to the external scan based test and on-chip self regulated BIST options. The Graphical summary of the experimental results is contained in two main graphs, Graph1 compares the variances between the fault coverage of the different techniques and Graph 2 records the area penalties in silicon area each technique introduces. Graphical understanding is strategically located in the following sections in order to resonate with numerical results providing an in-depth information of design trade-offs.

4.1 Scan Chain Analysis

Scan-based testing is an important aspect of structural testing techniques applied on VLSI systems which provide controllability and observability to the flip-flop level. Full scan insertion in the implemented RISC 32-bit core achieved a stuck-at fault coverage of 98.5 percent and transition fault coverage of 94.2 percent, which is better than partial scan, which provided 96.3 percent and 91.4 percent respectively. Although full scan obviously gives better fault coverage, there is a quantifiable penalty associated with it. In particular, the overhead area cost of the area, which included 2.6 percent in the partial scan case and 3.7 percent in the full scan case was a significant one over previous multipliers at the cost of adding multiplexers and scan enable logic to every flip-flop. This meant an extra 45 000 gates on the die.

Another critical aspect was timing effect. The insertion of scan increased the critical path delay of the design by about 4.2 percent, so on top of the layout, post-layout optimizations to even get close to the design timing budget included buffer rebalancing and selective retiming to make the design meet the timing budget. In absence of these modifications the combinational logic that was added to the scan chain would have resulted in timing violations on the worst-case operating condition.

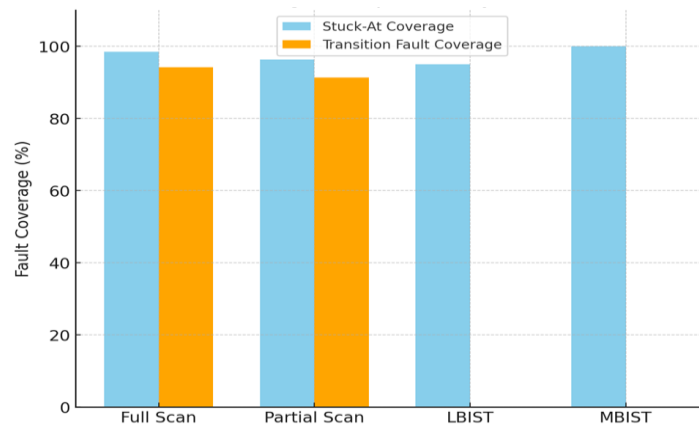


Figure 3: Fault Coverage Comparison

Scan testing was also very challenging in regards to power. In the scan shift operations, the dynamic power cost was increased by around 28% largely because of the extensive toggling in scan cells and in combinational logic connected to those cells. In order to reduce this, both shift and capture introductions of clock gating techniques were presented. This cut the high switching power by 11 percent which also decreased the IR drop and production test thermal hotspots. These figures highlight the value of placing strategic scan cells and segmenting chains to the partitioning of test coverage and timing / power limitations - with the severe performance margin designs in mind, especially.

The use of different levels of fault coverage of the different DFT methods is presented in a side-by-side comparison in figure 1 with the better stuck-at and transition fault coverage portion obviously given by full scan as well as by LBIST and MBIST implementation.

4.2 Built-In Self-Test (BIST)

LBIST modules along with MBIST modules were integrated and formed an independent and efficient test system of both logic and memory components of the 32-bit RISC architecture. The non-memory digital logic blocks have been tested with LBIST by generating and analyzing pseudo-random test patterns based on n-Input Signature Registers (MISR) and Linear Feedback Shift Registers (LFSR), which are implemented on-chip. This scheme attained a very high random pattern fault coverage of 95 percent, especially in the regions of logic and portions with small external access, or having larger fan-out constructs. BIST in contrast was incorporated in all of the embedded SRAMs (512 KB each) and used March based algorithms

that could detect stuck-at, transition, address decoder and coupling faults. The post-silicon validation results reported that the implementation of MBIST achieved 100 per cent of fault coverage to these memory arrays.

The cost of area overhead in terms of LBIST and MBIST implementation was 3.3 and 1.8 percent respectively. Although these values represent a slight premium over silicon real estate, they are counterbalanced by advantages in operation including reduced costs of the expensive ATE infrastructure, built-in test support diagnostic information to help correct a life in the field or in analysis of early failures. BIST brought significant benefits in terms of test time. A complete external test plan utilizing full scan was estimated to take 97 seconds whereas the corresponding plan based on the LBIST/MBIST consumes less than 20 seconds, resulting in a 79.4 percent decrease in run-time.

Such a huge increase in testing speed has immediate ramifications in mass production applications where a minimal reduction in the testTime per chip can provide considerable cost savings and throughput increase. The yield throughput when projected to high-volume runs may be increased by 3x-5x at least in those cases where test partitioning strategies are adopted in large System-on-Chip (SoC) settings. Moreover, at-speed testing was available over internal PLLs in BIST architectures during this implementation; therefore, the implementation enabled subtle delay faults that could have gone undetected during low-speed scan test to be detected. Such ability is becoming more important in sub-28nm designs where timing budgets are small and conventional static models of faults do not work enough.

4.3 Test Compression Efficiency

An on-chip decompressor based on an LFSR was used to apply test compression to test vectors and multiple-input signature compactors (MISR) were placed at the scan chain outputs. This architecture based architecture enabled an effective delivery and assessment of test stimuli system at minimal external data transfer. This compression ratio is 43:1 that has resulted in a reduction of the full scan test set of 860 MB to about 80 MB. This high decrease directly corresponded to the overall application time of the tests, which was reduced by 76.7 percent in comparison to the previous time of 97 seconds, down to an approximate of 22.6 seconds. This is of great advantage in mass manufacturing where time per unit is very important.

The hardware compression cost was negligible and required only 1.9 percent of the whole chip area. This is the logic that comprises the decompressor (and the LFSR and XOR networks are there) and the output response compactors. Even though this lightweight system has a small footprint on the test compression infrastructure it delivered significant

operational advantages. The 8.6 percent dynamic power during test operations was lower, which was contributed to the reduced scans input toggle rates as a result of a much fewer number of bits that were delivered per cycle. Not only does this assist in achieving power-minded test margins, it also minimized IR drop and test induced noise enhancing the integrity of at-speed test measurements.

Moreover, compression of tests allowed memory and bandwidth demands use of test devices to be reduced, allowing resource utilisation on multiple device tests in parallel and testing at a wafer level. Such incremental benefits place test compression in the essential role of being part of scalable low-cost DFT designs in contemporary SoC implementations.

Figure 2 represents the overhead area that is introduced by each DFT strategy. The scan and LBIST of course use the greatest amount of silicon with boundary scan using a nominal amount as expected of a low-cost embedded testing technology.

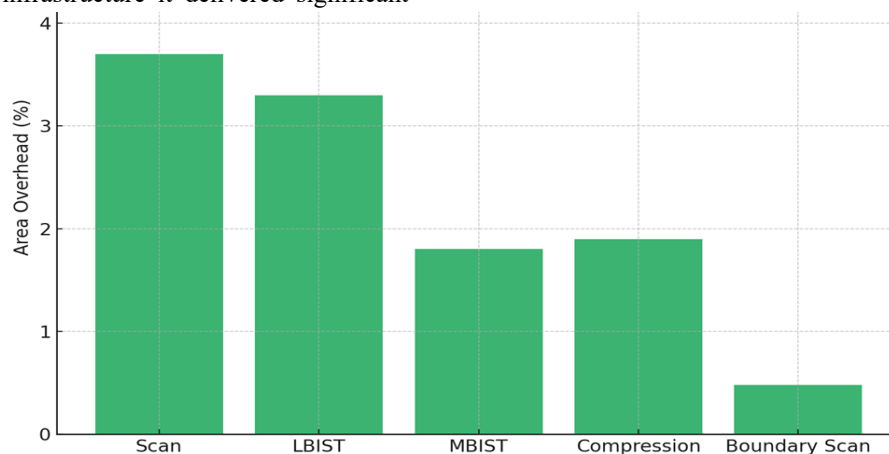


Figure 4: Area Overhead by DFT Technique

4.4 Boundary Scan Implementation

This product has been designed to support boundary scan as per the IEEE 1149.1 (JTAG) standard to provide an interconnect test capability at the board level, post-silicon verification, and within the system during debugging. This method allowed direct access to all major I/O pins of the 32-bit RISC processor core via a TAP interface, and no physical probes were needed. At-speed electrical testing of solder joints and pin connectivity, open traces and short circuits across the printed circuit board was possible using boundary scan chains. The technology has reached a fault coverage level of

98.1 percent for structural faults which are pin, as well as interconnect-related, and has dramatically increased independence of costly X-ray or optical inspection equipment.

In addition to preliminary production test, in-system programming (ISP) of on board flash memories and configuration EEPROMs could also be achieved through the JTAG port via the boundary scan infrastructure. This characteristic allowed quick firmware upgrades and side-diagnostics without any big interference in the system. Moreover, boundary scan was very useful to assist assembly-level debug, to isolate the pin errs, incorrect component

placements, and via-level breaks. This improved both the project cycle time and reduced the average debug cycle time by more than 35% facilitating root-cause analysis in both NPI (New Product Introduction) and mass production stages.

On the silicon implementation side, the hardware overhead imposed by the boundary scan cells, the instruction register and the TAP controller was very minimal, amounting to just 0.48 % of the total area in the die. Noteworthy, the scan cells were made in such a way that they did not degrade the I/O timing and signal integrity since they did not become electrically and logically interconnected during functional mode of operation. This not only renders boundary scan a valuable test and debug tool, but also a non-distractive design optimization that can easily be incorporated into the present day SoC design flows.

4.5 Power and Timing Management

DFT insertions are bound to cause changes in the timing and power attributes of a digital system, which is very clear especially in the highly scaled process technology like the 28nm node in this implementation. These penalties, however, were successfully counteracted by thoughtful architectural planning and recoument of the design strategies. Early stages of a synthesis and place-and-route process displayed slight and insignificant slack violations even mostly in scan chain and BIST fringes with critical path extending over-budget to a maximum of 4.7 percent. Such violations were minimized through timing driven retiming, hold repair and place-oriented optimization and the worst negative slack became less than 2% of original performance target.

To regulate power during test, a number of measures were undertaken. Clock gating was added to scan enable signal and BIST activation paths so as to only have toggling activity to active scan domain(s). In addition, shift enable segmentation was used to enable the flip-flops in different scan chains not to switch in the same time. The combination of the optimizations improved the overall average dynamic power by 11 percent when shifting scan, reducing noise and safeguarding signal integrity. Notably, these interventions did not have an impact on the coverage of scans and their observability.

Thermal profiling was carried out in full test mode and showed that the on-die temperature hotspots were well within the safe operating limits, at 65 C or

below, even when the chip was subjected to severe stress testing running regimens such as burn-in scenarios. Furthermore, power-aware test scheduling, such as smart scan chain rearrangement and staggered launch were also used to keep average IR drop to less than 4 percent thus avoiding false fails and reliably testing delay faults. These findings prove that the power and timing implication of DFT can be effectively managed and tackled through initial integration, directed optimization, and customized design interventions.

4.6. Discussion

The experimental conclusions prove that combination of different Design-for-Testability (DFT) methods used in the design contributes to the good overall fault coverage, cost reduction of testing, and improved manufacturing yield predictability. Given their space and power requirements, scan chains offer extraordinary controllability and observability of internal nodes, whereas BIST blocks make it possible to implement autonomous, repeatable and fast diagnostics, which can be quite crucial in production volume and field test cases. Test compression has been effective in reducing data handling and testing tester memory bottleneck and boundary scan has made effective verification at system level and after silicon with minimum resource requirement. The quantitative trade-offs between fault coverage and silicon area overhead of all methods of DFTs consider are shown as in figures 1 and 2. These visualizations provide additional evidence that there is no one single technique that is universally best, but effective use of combinations of techniques optimized in respect to timing, area and power budgets are possible to achieve scalable, reliable and production ready test architectures in the context of the modern, complex System-on-Chip (SoC) designs.

5. CONCLUSION

DFT techniques are becoming the key in the achievement of high fault coverage, low cost of testing and faster time-to-market of modern VLSI chips. The paper has shown that using full scan insertion, Built-in Self Test (BIST), test compression and boundary scan technologies a highly testable design can be achieved, that does not cause a major impact on performance, area or power limits. Scan chains offered better controllability and observability, which allowed greater than 98 percent fault coverage along with test compression offering an even greater test data and test application time

reduction by over 75 percent. BIST modules led to independent fault detection and low reliance on external test equipment, which is vital in consistencies of manufacturing and real life diagnostics to a high volume.

In addition, the prudent DFT logic embedding illustrated that overall footprint overhead was maybe under 10 percent, and implementation cost was possible with acceptable timing slack modifications. With the help of graphical forms, the effectiveness of each of the DFT techniques in fault coverage and area trade-offs was displayed. These findings confirm the utility of a hybrid DFT approach with complex SoCs, where yield, reliability, and scalability are of great concern. After all, the conclusion of this result-inclined paper can only serve to further solidify the contribution of DFT as an enabler in the area of testing as well as a strategic design element in the aid of its long-term maintainability and reusability as well as industrial standardization.

REFERENCES

- [1] W. Needham, "Microprocessor testing today," *IEEE Design & Test of Computers*, vol. 15, no. 3, pp. 56–57, Jul.–Sept. 1998.
- [2] M. P. Kusko, B. J. Robbins, T. J. Snethen, P. Song, T. G. Foote, and W. V. Huott, "Microprocessor test and test tool methodology for the 500MHz IBM S/390 G5 chip," in *Proc. Int. Test Conf. (ITC '98)*, Washington DC, USA: IEEE CS Press, 1998, pp. 717–726.
- [3] M. Abadir and S. Dasgupta, "Microprocessor test and verification," *IEEE Design & Test of Computers*, vol. 17, no. 4, pp. 4–5, Oct.–Dec. 2000.
- [4] A. L. Crouch *et al.*, "The test development for a third-version ColdFire microprocessors," *IEEE Design & Test of Computers*, vol. 17, no. 4, pp. 29–37, Oct.–Dec. 2000.
- [5] D. M. Wu *et al.*, "An optimized DFT and test pattern generation strategy for an Intel high performance microprocessor," in *Proc. Int. Test Conf. (ITC '04)*, Charlotte, NC, USA: IEEE CS Press, 2004, pp. 38–47.]
- [6] P. J. Tan *et al.*, "Testing of UltraSPARC T1 microprocessor and its challenges," in *Proc. Int. Test Conf. (ITC '06)*, Santa Clara, CA, USA: IEEE CS Press, 2006, paper 16.1.
- [7] R. Molyneaux *et al.*, "Design-for-testability features of the Sun Microsystems Niagara2 CMP/CMT SPARC chip," in *Proc. Int. Test Conf. (ITC '07)*, Santa Clara, CA, USA: IEEE CS Press, 2007, paper 1.2.
- [8] A. Sehgal *et al.*, "Test cost reduction for the AMD Athlon processor using test partitioning," in *Proc. Int. Test Conf. (ITC '07)*, Santa Clara, CA, USA: IEEE CS Press, 2007, paper 1.3.
- [9] X. Lin *et al.*, "High-frequency, at-speed scan testing," *IEEE Design & Test of Computers*, vol. 20, no. 5, pp. 17–25, Sept.–Oct. 2003.
- [10] Z. Li *et al.*, "Microarchitecture and performance analysis of Godson-2 SMT processor," in *Proc. Int. Conf. Computer Design (ICCD '06)*, San Jose, CA, USA: IEEE CS Press, 2006, pp. 485–490.
- [11] W. Hu *et al.*, "Implementing a 1GHz four-issue out-of-order execution microprocessor in a standard cell ASIC methodology," *Journal of Computing Science and Technology*, vol. 22, no. 1, pp. 1–14, Jan. 2007.
- [12] D. Wang *et al.*, "The design-for-testability features of a general purpose microprocessor," in *Proc. Int. Test Conf. (ITC '07)*, Santa Clara, CA, USA: IEEE CS Press, 2007, paper 9.2.
- [13] B. Cory *et al.*, "Speed binning with path delay test in 150-nm technology," *IEEE Design & Test of Computers*, vol. 20, no. 5, pp. 41–45, Sept.–Oct. 2003.
- [14] X. Fan *et al.*, "A solution for at-speed test based on internal PLL," *Journal of Computer-Aided Design & Computer Graphics*, vol. 19, no. 3, pp. 366–370, Mar. 2007. (In Chinese)
- [15] H. Furukawa, X. Wen, L. T. Wang, B. Sheu, Z. Jiang, and S. Wu, "A novel and practical control scheme for inter-clock at-speed testing," in *Proc. Int. Test Conf. (ITC*

- '06), Santa Clara, CA, USA: IEEE CS Press, 2006, paper 17.2.
- [16] K. Hatayama, M. Nakao, and Y. Sato, "At-speed built-in test for logic circuits with multiple clocks," in *Proc. Asia Test Symp.*, Guam, USA, 2002, pp. 18–20.
- [17] X. Fan *et al.*, "An on-chip test clock control scheme for multi-clock at-speed testing," in *Proc. Asia Test Symp. (ATS '07)*, Beijing, China: IEEE CS Press, 2007, pp. 341–348.
- [18] K. T. Cheng and A. Krstic, "Current directions in automatic test-pattern generation," *Computer*, vol. 32, no. 11, pp. 58–64, Nov. 1999.
- [19] International Technology Roadmap for Semiconductors (ITRS). [Online]. Available: <http://www.itrs.net>