A Structured Design for Test Methodology

Kumar Venkat

Silicon Graphics, Inc.

2011 N. Shoreline Blvd., Mountain View, CA 94039

Abstract

This paper is a case study of a structured Design for Test (DFT) methodology that was formulated for a major system design project consisting of 11 complex ASICs. The methodology includes full scan for chip test, and an optimized boundary scan for board test. The paper discusses details of the ASIC designs and technology, the DFT methodology, the design of test logic, ATPG tool selection, development of in-house tools, and integration of DFT into the overall design flow. This project has demonstrated that DFT can be considered early in the design and integrated efficiently into the design flow.

1. Introduction

This paper describes a structured Design for Test (DFT) methodology that was formulated and used in a major system design project at Silicon Graphics. The goal was to integrate DFT into the overall ASIC design methodology and establish a uniform DFT process for all the ASICs to follow. The methodology, tools, and the design of test logic were common to all ASICs in the project. The paper discusses details of the ASIC designs and technology, the DFT methodology, the design of test logic, ATPG tool selection, development of in-house tools, and the integration of DFT into the overall design flow.

The project consisted of the design of eleven complex ASICs. The ASICs were all in the complexity range of 50,000 to 100,000 utilized gates, operating at 50 MHz or above. The technology used was LSI Logic's LCA200K/LCA210K gate array with 0.7 micron feature sizes, using two or three metal layers for interconnection. Die sizes ranged from 10mmX10mm to 15mmX15mm. Several designs included compiled on-chip RAMs. Some designs included synchronizers, multiple system clocks on the same chip, and asynchronous inputs. Some of the chips were datapath chips with a fairly regular structure including registers, FIFOs, parity logic, error correction logic, etc. Others were complex control-oriented chips with a large number of state machines and very little regularity of structure.

The intent of this paper is to present a practical test-driven design methodology from a large design project in the computer industry, and evaluate the success of the approach and some of the results obtained. After addressing the motivation and the need for design for test, the DFT goals that were set for the project are described. DFT guidelines that were proposed and implemented for chip test are discussed in detail, including how they were integrated into the design methodology. DFT for board interconnect test is then described, particularly the optimizations for speed. Finally, development of scan synthesis software and selection of an ATPG tool are discussed, followed by a summary of fault coverage results.

2. Motivation for DFT

High-volume product lines require a high level of confidence in the components before final system assembly. Most system manufacturers find it too expensive and difficult to completely test an assembled system on the manufacturing floor. It is also virtually impossible to measure test coverage of the functional system tests used in manufacturing. Even if systems could be tested exhaustively, it is a major challenge to identify and replace faulty boards or components. Therefore, as chips and boards get increasingly complex, adequate testing of the chips and boards as components has become a prerequisite to successfully integrate a high-volume system product.

While it is easier to test the components stand-alone before system assembly, the cost of a defective component that enters the manufacturing process can be high and will increase in proportion to how late in the process the component is identified and replaced. If the limited tests in manufacturing do not identify the defective component, the system may be shipped to a customer, in which case the cost of the defect can be very high indeed when the component finally exhibits the failure mechanism. The manufacturing process thus places the bulk of the test burden at the component level and relies on high quality component tests for overall success.

Clearly, the components (both chips and boards) need to be highly testable in order for the system to be manufacturable in volume. However, there is no simple way to determine what type of fault models should be considered for the chips and what level of fault coverage is acceptable [2].

Let us consider the following classical model for defect level [1]:

$$\text{Defect Level} = 1 - Y^{(1-FC)} \qquad (1)$$

where Y = process yield, and FC = fault coverage.

Starting with a preliminary defect level criterion, one faces a number of problems. First, process yield Y is hard to measure and hard to control, and is very uncertain for new technologies. Second, for a given process yield and technology, it is difficult to say what kind of physical defect modes are important and what type of fault models (stuck-at, bridging, delay) are necessary to model them adequately. Third, for a given set of fault models of interest, capabilities of ATPG tools and capabilities of testers play a critical role in determining which fault models are finally selected for coverage. Finally, the additional design-for-test effort and area/speed penalties of each test method must be considered along with quality requirements.

Based on both the general capabilities of commercial ATPG tools available in 1991 and the tester capabilities available to the project at that time, it was decided to rely on the single stuck-at fault model. Assuming a process yield of 70%, fault coverage in excess of 97% would be required to keep the defect level under 10000 ppm. This was an informal fault coverage goal for the ASICs.

Given the wide variation in the complexity of the chips and possible variations in the design styles of various designers, it was clear that a structured design for test methodology was required to meet the fault coverage goal on all the chips within a reasonable amount of time.

In addition, board interconnect testing needed to be addressed because of difficulties stemming from high-density PCBs, lack of access to internal nets on the PCB, and new chip packaging technologies that do not allow access to ASIC pins [4]. It was decided to include DFT for board interconnect test and integrate it into the overall chip design effort.

3. DFT Goals

The following overall DFT goals were established for the project.

For chip test, full scan and partial scan were two possible strategies. It was also important that test generation be completed in a few days without requiring iterations or modifications to the design. Full internal scan was selected in order to simplify the test problem and meet the overall project goals. This also provided a larger pool of ATPG tools to select from, for combinational test generation. The DFT guidelines (discussed in section 5) were designed explicitly to improve testability for a combinational ATPG tool. Further, parts of a chip that were not testable using combinational ATPG, such as RAMs and synchronizers, were required to have easy access from the I/O pins for testing with functional vectors.

For board test, it was desirable to use the IEEE 1149.1 boundary scan standard. However, due to difficulties in meeting I/O timing, a timing-optimized subset of this standard was adopted as described in section 6. The implementation was still required to be "JTAG-compliant" so that ATE software could generate test vectors for board interconnect test.

4. Design Methodology

The overall design methodology is test-driven in that testability considerations influence every major step of the design process. This section describes the methodology in terms of the test-related activities in the design.

The top level of each chip is partitioned into four major components: *frame, core, scan control, clock generation*.

The frame contains all the I/O buffers and pad registers for inputs, outputs and output-enable signals. The core contains all the internal logic other than the I/O logic.

The scan control module includes all the test control logic. Boundary scan is based on the IEEE 1149.1 standard (JTAG) with some modifications to optimize I/O. Internal scan is based on direct external control, bypassing the JTAG test access port (TAP) controller. In boundary scan mode, the scan chain in the frame module is selected. In internal scan mode, the scan chains in the frame and the core are concatenated to form a single internal scan chain.

The scan control module includes the reset logic for the chip, making sure that flip-flops are not reset asynchronously while in scan mode. It also controls bypassing of the on-chip phase-locked loop (PLL) with a test clock within the clock generation module. It generates most control signals with sufficient hold-time margin (typically half a cycle) to ensure that clock skew in the test clock network does not pose a problem.

Many of the DFT guidelines are applied at the level of the RTL design to ensure that the design would be highly testable. Most of the guidelines are very fundamental for achieving high testability and can not be easily applied at a later stage in the design cycle. All the key DFT guidelines developed for the project are discussed in sections 5 and 6.

Logic synthesis is used to enforce the choice of flip-flops and other storage elements used in the design. A selected list of flip-flops is used in synthesis such that all of them could be replaced with scan equivalents at a later stage. Pre-scan timing analysis and optimization are performed at this stage, with a 5% (1 ns) allowance for overhead from scan. Area estimates and optimizations also consider an overhead from scan of about 10 to 15%.

Scan synthesis is performed on the synthesized netlist. Scan synthesis converts all the flip-flops into scan equivalents, and connects the scan chains in the core and frame modules separately. The structure of the full-chip scan chain is controlled by the scan control module as described above.

5. DFT Guidelines for Chip Test

The DFT guidelines for ASIC design covered a number of potentially troublesome areas which could lead to loss in test coverage from scan vectors [2]. Given the large number of new ASICs being designed, variation in design complexity and variation in design styles, these guidelines served to standardize the solutions to similar types testability problems across different designs. Some of the key guidelines that were implemented are described below.

Clocking and reset: In scan test mode, the on-chip PLL was bypassed with a test clock which was controllable directly from an input pin. In addition, clock gating was not allowed except for generating write pulses to on-chip RAMs. The reset signals were forced to be inactive in scan mode since asynchronously resettable flip-flops were needed in the design. Further, any flip-flop in the design resetting other flip-flops asynchronously was prohibited, at least for the scan mode.

Choice of flip-flops: Rising-edge flip-flops were recommended for use in all of the designs. A list of acceptable flip-flops was used which all had equivalent scan flip-flops in the library. Use of falling-edge flip-flops and latches were minimized as much as possible. In a few instances where they needed to be used, it was recommended that latches be forced to be transparent and falling-edge flip-flops be forced to clock freely in scan test mode. An additional restriction in scan test mode was that falling-edge flip-flops could not be cascaded so that each instance of it could be modeled as a buffer in the midst of rising-edge flip-flops.

Multiplexers and tristate buffers: Selection of multiplexer inputs or tristate buffers had to be made mutually exclusive at all times, in order to avoid electrical conflicts during scan shift operations.

On-chip RAMs: As mentioned earlier, RAMs could only be tested using functional vectors. This required reasonably easy access to the RAMs from the I/O pins. Further, in order to prevent the RAMs from adversely impacting the testability of surrounding logic, additional registers were recommended in some cases to observe inputs to the RAMs (such as address, data, control) and an output multiplexer was recommended for controllability of the RAM output. Fig. 1 illustrates the scheme.

Multiple system clocks: Some of the designs required two system clocks for normal operation in the system. To make the test problem simpler, a single scan chain was recommended for these designs. In such cases, two clock generation modules were included at the top level of the chip, and the same test clock was used to bypass both PLLs. Synchronizers were bypassed in scan mode, with added delay in the bypass paths for hold time. Fig. 2 shows this.
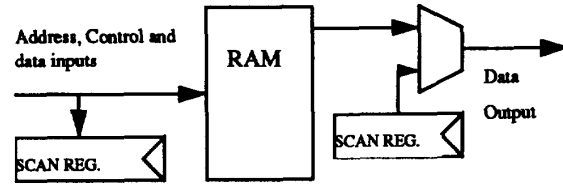

Fig. 1. Testability enhancement surrounding a RAM block.

6. DFT for Board Test

A subset of IEEE 1149.1 [3] was implemented to support board interconnect testing while optimizing the I/O timing as much as possible. The functional pad registers associated with all inputs and outputs were utilized for the boundary scan chain as well. Each input, output or output-enable signal had a pad register in the frame module of the chip. This register was then converted to a scan equivalent and served as the shift/capture register for boundary scan. This eliminated the need for dedicated boundary-scan registers and avoided the resulting multiplexer delays in I/O paths.

Since the boundary scan registers were in capture mode by default, the test clock (applied to the system clock network by bypassing the PLL) was allowed to run only in the DR_Shift and DR_Capture states of the TAP controller, so that data would be retained in the shift/capture registers in all other states.

Apart from the basic bypass and device identification instructions, only the EXTEST instruction was implemented for interconnect test. Fig. 3 shows the structure of a shared output scannable pad register. In most cases, update registers were not used with outputs in order to save the multiplexer delay. However, output-enables required update registers so as to avoid bus contention during boundary scan shift operation. Update registers essentially "hide" the data shifting occurring on the chip. In all cases, the guiding principle was to avoid direct or indirect bus contention on the board during shift operations. The limited number of update registers were clocked specifically in the DR_Update state to transfer data from shift/capture registers.

Sharing of the boundary-scan register with the functional pad register and elimination of the update register resemble some of the features of the proposed IEEE P1149.2 standard [5].
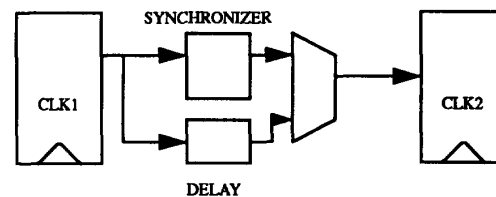

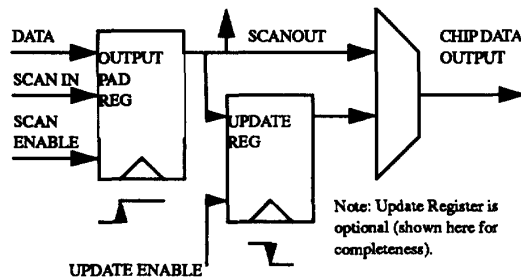Fig. 2. Bypassing synchronizers with added delay.

Fig. 3. Output functional pad register utilized for scan.

Another area that required special care involved asynchronous inputs that were: (a) registered at the input pads using a special external clock, or (b) synchronized to the internal clock using a synchronizer. In these cases, adherence to boundary scan principles required adding dedicated boundary-scan input registers for testability, and leaving the functional registers out of the scan chain. For internal scan, the synchronizers were bypassed as described in section 5.

7. Tools

An in-house tool *scansyn* was developed to address the scan synthesis requirements of the project. Many of the requirements were driven by physical design issues such as minimizing routing area, minimizing timing penalties, and ensuring correct scan operation without race conditions in the presence of clock skew.

Some of the key features of *scansyn* include: (1) Hierarchical scan insertion which follows the chip floorplan at the block levels. (2) Buffering of global scan control signals within each module. (3) "Black-box" option to leave selected modules out of the scan chain. (4) Optional added delay in the scan path for improved hold-time in the presence of large clock skews. (5) Use of any unused flip-flop output (Q or QN) for scan propagation, with a final correction in case of a net inversion in the scan chain, to minimize loading on flip-flops. (6) Buffering of scan outputs from each module to avoid large capacitive loading on the last flip-flop in the module due to wiring.

An ATPG tool, the *Sunrise Testgen*, was selected for the project based on a number of important considerations: (1) Fault coverage in the high 90's using combinational ATPG on actual test cases from the project. (2) Test generation time, on large designs, of less than 36 hours. (3) Size and compactness of vector sets. (4) Foundry interface in terms of netlist and vector formats. (5) Design rule checking to aid designers in identifying testability problems. (6) Support for inversions in the scan chain.

8. Current Status and Results

All of the ASIC designs have been completed at this time and fabricated chips have been tested successfully using scan

Design Name	No. of Faults	No. of Flip-flops	No. of Scan Vectors	Fault Coverage with Scan
C1	47734	2622	5285	96.3
C2	93874	3613	8700	99.3
C3	74473	4031	2687	97.3
C4	29895	1322	1571	98.73
C5	20091	749	679	97.0
C6	67420	3010	3367	99.4
C7	24693	1903	598	96.0
C8	51325	3168	641	94.3
C9	71656	3179	1255	90.0
C10	32234	2326	582	99.82
C11	39654	2626	926	99.11

vectors. Table I summarizes the fault coverage results obtained. Fault coverage for functional vectors is not included. On-chip RAMs are excluded from these statistics.

A few designs did not completely conform to the DFT guidelines, particularly in the way that RAMs and synchronizers were bypassed in scan mode. This has resulted in additional untestable faults in those chips, which have to be covered by functional vectors. These deviations from guidelines were mostly a result of area and/or timing constraints, and they highlight the cost of testability incurred in many other cases where the guidelines were closely followed.

9. Conclusion

This paper has described a structured DFT methodology, and integration of test considerations and tools into the overall design flow in a large design project. The methodology has allowed the project to meet most of the stated test goals. The design process will allow high-volume manufacturing of the product with a good degree of confidence in the quality of the components. The methodology has also minimized timing degradation of the designs through intelligent scan synthesis.

References

[1] T.W. Williams and N.C. Brown, "Defect Level as a Function of Fault Coverage", IEEE Transactions on Computers, Vol. C-30, No. 12, December 1981, pp. 981-988.

[2] K. Venkat, "Integrating Test into ASIC Design", EDN, in press.

[3] IEEE Standard Test Access Port and Boundary Scan Architecture (IEEE STD 1149.1), 1991.

[4] S. Broderick and K. Wills, "Intricacies of Boundary Scan Solutions", Evaluation Engineering, September 1991.

[5] B. Dervisoglu, "Boundary-Scan Update: IEEE P1149.2 Description and Status Report", IEEE Design and Test of Computers, September 1992, pp. 79-81.