Test Methodology for Freescale's High Performance e600 Core Based on PowerPC® Instruction Set Architecture

Nandu Tendolkar, Dawit Belete, Ashu Razdan, Hereman Reyes, Bill Schwarz, Marie Sullivan
Freescale Semiconductor, Somerset Design Center
7700, West Parmer Ln.
Austin, TX 78729
Nandu.Tendolkar@freescale.com

Abstract

This paper presents the DFT techniques used in Freescale's high performance e600 core. Highlights of the DFT features are at-speed logic built-in self-test (LBIST) for delay fault detection, very high test coverage for scan based at-speed deterministic delay-fault test patterns, 100% BIST for embedded memory arrays and 98% stuck-at fault test coverage for deterministic scan test patterns. A salient design feature is the isolation ring that facilitates testing of the core when it is integrated in an SoC or host processor.

1.0 Introduction

Freescale's e600 is an enhanced version of the G4 core used in the high-performance MPC74xx family of host processors containing PowerPC® cores. It is planned to scale beyond 2 GHz and to support chip multiprocessing (CMP) while maintaining full compatibility with the PowerPC instruction set architecture. Like its G4 predecessor, the superscalar e600 core is designed to issue four instructions per clock cycle (three instructions plus one branch) into eleven independent execution units, and to include a full 128-bit implementation of Freescale's advanced AltiVec Single Instruction Multiple Data (SIMD) vector processing technology. The e600 has a 1 Megabyte Level 2 cache with error correction capability. A block diagram of e600 is shown in Figure 1.

e600 is the first G4 core to be implemented in 90 nanometer CMOS SOI technology. It is 45.85 mm square. It is designed to run at up to 2 GHz frequency.

The focus of this paper is the innovative DFT features of the e600 core.

2.0 Overview of the e600 core DFT Advances

The e600 core shares many DFT features with Freescale's MPC74xx [1,2] and the e500[3]core. Several papers have been published on DFT of MPC74xx [1,2,4,5,6]. Like MPC74xx, the e600 core's DFT features include full-scan design (LSSD), BIST for embedded memory arrays (ABIST), logic BIST for stuck-at fault testing, and scan based at-speed delay fault testing. These proven DFT techniques result in very high scan based stuck-at fault test coverage for logic and 100% test coverage for faults in array bit cells.

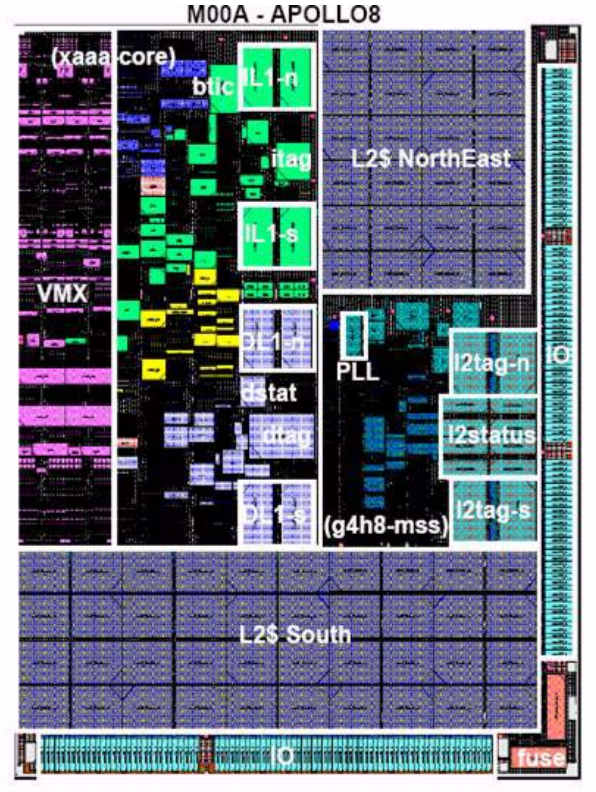

Figure 1: e600 Core Block Diagram

Building from that, however, several DFT advances were made on the e600 core. Like the e500, the e600 core also has unique DFT features to support core testing when the core is integrated on an SoC. See Reference [3] for details of the DFT features of the e500 core. The main components of the test strategy are:

- Design for core testability in an SoC environment
- Scan patterns for both stuck-at and at-speed transition and path delay faults

0-7803-9039-3/$20.00 © 2005 IEEE

- Logic BIST (LBIST) capability that supports both slow speed and at-speed tests
- Array Built-in self test (ABIST) for all memory arrays
- Core test patterns are reused for SoC testing.
- All test patterns are verified using Verilog before delivery to improve test pattern quality.

Table 1 contains some basic facts about the e600 core that are relevant to testing.

Number of LSSD Latches	87k
Number of Scan Chains	76
Length of the longest scan chain	876
Number latches in core wrapper	808
Stuck-at Fault Universe	~8.6 million

Table 1. e600 core Design Statistics

The starting point for DFT design of the e600 core was the MPC7447A DFT design. The DFT features of MPC7447A are described in [1]. Since the e600 is a core, DFT features were added to support core testability. The testability goals for the e600 core are to be as good as MPC7447A in DC stuck-at fault test coverage and array BIST (ABIST) and to have significantly improved transition fault test coverage. The DC stuck-at fault test coverage of MPC7447A is over 98%. Using the same proven DFT techniques in the e600 core, we expect that the DC stuck-at fault test coverage of the e600 core would be 98%. One area where improvement was needed was at-speed delay fault testing. Whereas the transition fault test coverage for MPC7447A is 75%, the transition fault test coverage goal for the e600 core is 90%. We added several DFT features in the e600 core to increase the transition fault test coverage. MPC7447A has LBIST for stuck-at fault detection. In the e600 core, we implemented at-speed LBIST to detect delay defects in addition to stuck-at faults.

The following novel DFT features of the e600 core are discussed in this paper:

- **Test Partitions:** We use test partitions to reduce tester memory and test time.
- **DFT for Core Test:** Facilitates testing of the core when it is embedded in a chip (SoC).
- **Deterministic Scan (LSSD) based at-speed testing:** The e600 core provides support for at-speed delay fault testing while allowing arbitrarily slow shifting of scan chains. We discuss features to support increased test coverage goals of the e600 core. These features are not in the MPC74xx microprocessors.
- **At-speed LBIST:** In the e600 core, the LBIST runs at core speed and can be invoked when testing a chip on a tester or during power-on-reset. Previous MPC74xx microprocessors supported only stuck-at fault LBIST test. Additional LBIST features in the e600 core reduce the test time by a factor of 2.
- **Custom Array Testability:**

The e600 core contains a large number of custom memory arrays and our strategy is to specify a set of array design rules that are necessary to get high test coverage using scan test patterns. The custom array designers follow these rules. We use an ATPG tools to check that the custom array design rules are followed. We also do a test coverage analysis of each custom array before the design is released.

The remainder of the paper is organized as follows. The DFT features are presented in sections 3 through 7. Section 8 contains a discussion of custom array testability. Section 9 contains discussion and conclusions.

3.0 Test Partitions and ATPG Models

The limited availability of pins puts a limit on the maximum number of external scan chains that can be used for scan based tests. The e600 core has 76 external scan chains that operate in parallel. Test partitions allow us to test a subsection of e600 using a subset of the scan latches. Partitioning allows us to reduce the length of the longest scan chain and save tester memory. For testing a partition, the relevant scan cells are stitched into 76 internal scan chains. This provides an efficient way of testing the partition without having to load (using scan) all latches that exist in the core. Like the MPC7447A microprocessor, the e600 core has two partitions. For detecting a small set of faults, loading and unloading of latches in both partitions is required. From test perspective the e600 core is divided into three zones:

1. Zone 1: This contains the Level 1 caches and the instruction processing circuits. Approximately 77% of the total faults belong to this unit. It has 58326 latches. The longest scan chain has 876 latches during scan tests. For DC and AC scan tests, it has 71 scan chains.

2. Zone 2: Zone 2 contains the memory subsystem (MSS) and the core isolation ring. It has a 1 MB Level 2 cache and the system bus interface. Zone 2 has 70 scan chains. The total number of latches in Zone 2 is 26680. The longest scan chain has 408 latches. Approximately 21% of the total faults belong to this unit.

3. Zone 3: Zone 3 has the test control, clock control and the debug functions. Zone 3 has 5 scan chains and has 1629 latches. It has approximately 2% of the total faults.

In the e600 core, the clocks and the scan chains of each zone can be independently controlled during test. Scan chain muxing is shown in Figure 2.

The following three models are used for generating scan test patterns for stuck-at and transition faults:

Model I: This model has Zone 1 and Zone 3. Zone 2 is black boxed. Its clocks are turned off. Seventy-six scan chains are loaded and unloaded in parallel.

Model II: This model has Zone 2 and Zone 3. Zone 1 is black boxed. Its clocks are turned off. Seventy-five scan chains are loaded and unloaded in parallel.

Model III: This is the full core model, with all zones present. It requires two separate load operations and unload operations to load and unload the scan chains. The Zone 1 and Zone 3 latches are loaded first. During this time, clocks to Zone 2 are off. Then the clocks to Zone 1 and Zone 3 are turned off and the clocks to Zone 2 are enabled. Thus latches for Zone 2 are loaded using scan without disturbing any of the latches in Zones 1 and 3. Test patterns are generated using model III for only those faults that are not detected by using models I and II. A single test pattern for Model III requires 1284 bits of tester memory and 1284 shift clock cycles for loading and unloading.

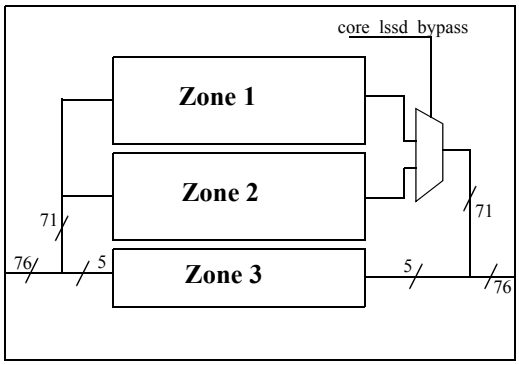

Figure 2: Scan Chain Muxing

We want to note that, in the MPC7447A microprocessor, two of the three zones are clocked during scan, Zone 3 and either Zone 1 or Zone 2 Thus in Mode III, the Zone 3 latches cannot be used as control and observe points. Since Zone 3 has latches that control the at-speed clock sequence, Mode III model could not be used for generating AC Scan test patterns for the MPC7447A microprocessor. In the e600 core, all three zones can be scanned in Model III, which allows us to generate transition fault test patterns in Mode III to improve test coverage.

4.0 DFT for Core Testing

In this section we describe the DFT features of the e600 core that facilitate testing it in a SOC. Reference[3] has a description of the core test interface for Freescale's e500 core. References [7,8,9] discuss various aspects of core testing and core test interface. We focus on how the core testing issues were handled in the e600 core. We discuss the core isolation ring concepts below.

4.1 Core Isolation ring

Core isolation ring's function is similar to that of a wrapper used in SoC core testing[7,8,9]. The salient features of the e600 Core Isolation Ring are:

- It improves testability of the e600 chip during various SOC testing modes.
- The test data used to test the core does not depend on the SOC that uses the core.
- It adds minimal delay to the functional path.
- During functional operation, it provides the ability to capture data that can be used for silicon debug.
- It can be controlled to drive out safe values out of the core and drive safe values into the core during Power On Reset operation.

The e600 core isolation ring is shown in Figure 3. It contains the following:

- A cell for each non-test control input and output port of the core. The contents of a cell are shown in Figure 4. Each cell contains two scannable latches. The core isolation ring latches are connected to form two scan chains during scan testing and LBIST. For at-speed delay fault testing, each cell that controls a core input has the capability to launch a at-speed transition. Each cell that controls a core output pin captures a at-speed transition.
- Buffering for test control signals

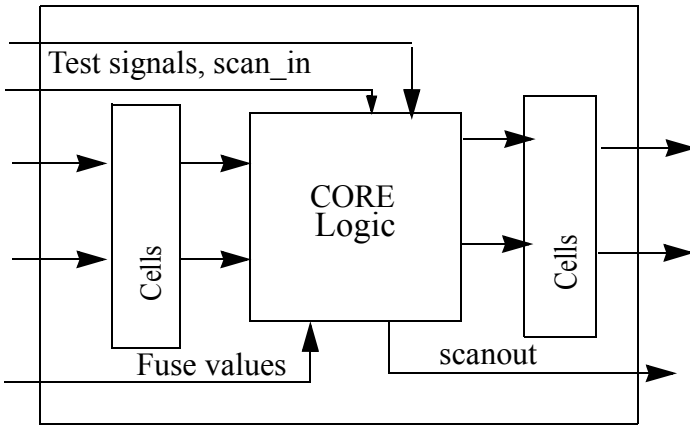

Figure 3: Core isolation ring

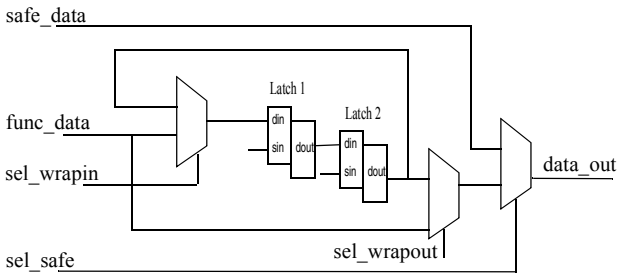

Figure 4: Isolation ring cell

4.2 Modes of Operation

The function of the core isolation ring depends on the mode of operation. The control logic configures the core isolation ring according to the mode that the user has selected. Table 2 below describes its functionality during the various modes of operation.

In Table 2, the first row refers to scan tests. The word wrapped refers to the following: on the input side, the core input comes from the core isolation ring cell for the input; on the output side, the core output is driven from the core isolation ring cell for the output. In functional mode of operation, platform signals drive the core input and core outputs drive platform signals.

CONTEXT	e600 Inputs	e600 Outputs
Scan Test (DC/AC)	WRAPPED	SAFE
LBIST (DC/AC)	WRAPPED	SAFE
Platform Test (DC/AC)	SAFE	WRAPPED
FUNCTIONAL	FUNC (transparent)	FUNC (transparent)
DEBUG	SAFE/FUNC/ WRAPPED	SAFE/FUNC/ WRAPPED
POR I	FUNC	FUNC
POR II	SAFE	SAFE

Table 2. Modes of operation

There are two ways of performing power on reset (POR). For each pin, a specific value (Vdd or Ground) is defined as the SAFE value.

4.3 Clocking of Core Isolation Ring

The core isolation ring employs a unique clocking scheme to provide flexibility during the various test and debug modes. This clocking scheme allows it to function in three clocking modes: the core isolation ring clocks can run a) on the high-speed internal system clock (GCLK) for at-speed transition fault testing, b) the slow external test clock for stuck-at fault testing, c) the JTAG test clock (TCK) for silicon bringup/debug. Figure 5 below shows the clocking circuitry that allows the core isolation ring to achieve these clocking modes.

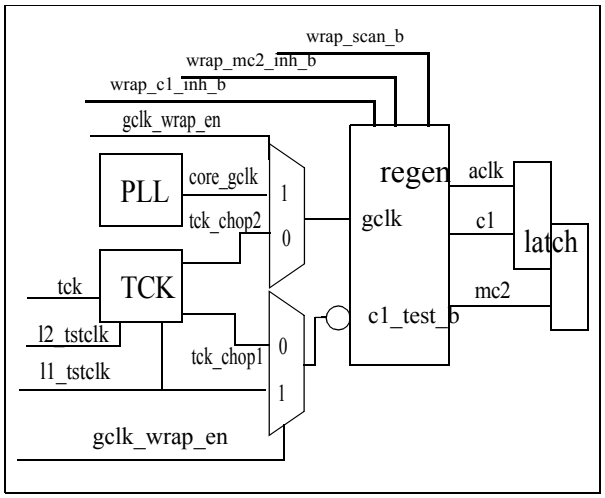

Figure 5: Clocking of isolation ring

The primary component of the core isolation ring clocking circuitry is the clock regenerator module (denoted as *regen*) in Figure 5 above. This module receives input source clocks (shown left of the *regen*) and based on the control signals (shown above the *regen*) produces the appropriate output clocks that feed the core isolation ring latches.

5.0 LSSD Scan Test Modes

Table 3 shows the test modes available for scan based tests. Three core input signals are used to select a specific test mode.Two major types of scan based tests are DC or stuck-at fault tests and at-speed delay fault or AC scan tests. The signal core_lssd_test_modes[0] selects the DC or AC mode of testing. Both DC and AC scan modes have four submodes as described below.

LSSD test mode [0]	LSSD test mode [1]	LSSD test mode [2]	*lssd_bpyass*	*ary_frz*	*slow_ac*	*ac_chains*
0	0	0	-	-	+	-
0	0	1	-	+	+	-
0	1	0	+	-	+	+
0	1	1	+	+	+	+
1	0	0	+	-	-	+
1	0	1	+	-	+	+
1	1	0	+	+	-	+
1	1	1	+	+	+	+

Note: “+” indicates asserted; “-” indicates deasserted.

Table 3. Test modes

Test Mode Description:

- lssd_bypass - Assertion of this signal re-configures the scan chains to exclude certain latches that are expected to have 1-hot or all off values.
- ary_frz - Assertion of this signal disables the clocks to the custom array circuits. Array output is the precharge value.
- slow_ac - During at-speed transition fault testing, this signal disables logic that is known to be non-functional.
- ac_chains - Assertion of this signal re-configures a scan chain so that latches impacting PLL lock are excluded.

6.0 Logic BIST

In this section we discuss the salient features of e600 LBIST. As discussed earlier, starting from the MC7447A microprocessor LBIST implementation, several improvements have been made in the e600 core. In the e600 core, the LBIST has both slow speed and at-speed clocking. The LBIST engine supports a larger number of clocking sequences too. This allows us to use LBIST for delay fault and stuck-at fault detection. The MPC74xx microprocessors did not have at-speed LBIST. In the e600 core, the number of scan chains in the core is doubled to reduce the length of the longest scan chain during LBIST to 440 latches.Changes were made to the LBIST design to allow for simultaneous testing of both Core and MSS.

MPC7447A LBIST design of the MPC7447A microprocessor is discussed in Reference 1. Reference [3] contains the description of the LBIST implementation of Freescale’s e500 core. Reference[10] contains the STUMPS architecture that is the basis of many LBIST implementations. References[11,12] are examples of how LBIST has been used in large industrial designs.

In the e600 core, an 71 bit LFSR-based PRPG (Pseudo-Random Pattern Generator) register is used to generate pseudo-random bit-sequences, which are shifted into the scan chains. A programmable sequence of functional clocks are issued to capture the test response. The scan chains are then shifted into a 71 bit LFSR-based MISR (Multiple-Input Signature Register), which calculates a signature[10].

6.1 At-Speed Clocking

The clocking sequences supported for at-speed clocking during LBIST are same as those for the scan based at-speed testing. Anywhere from zero to eight C1/C2 clock pairs can be used; either a C1 or a C2 can be used to start or end the sequence. At-speed clocks are generated using the same high-speed test clock engine that is used for AC scan.

During LBIST, an at-speed clock burst is generated automatically as soon as the engine finishes the scan load of a pattern. Clock control passes from the LBIST engine to the AC test clock engine, which generates the specified number of high-speed clocks from the PLL. A simple shift register is used to count the clocks. Control then passes back to the LBIST engine so that scan unload can begin. The test is entirely self-contained and uses the same clock controls and distribution that are used functionally.

6.2 Test-Time Improvements

One of the main concerns with using LBIST for previous MPC74xx microprocessor designs was the test time required to run LBIST. The LBIST for MPC74xx microprocessor required running two tests. In the first test, the Core was tested. The test time for this test is proportional to the the length of the longest scan chain. The core had 71 scan chains and the longest scan chain had 876 latches. In the second test the MSS was tested. The MSS had 70 scan chains and the longest scan chain had 404 latches. In the e600 core, LBIST design allows simultaneous testing of both the Core and the MSS. Further, there are 142 total scan chains for Core and 70 scan chains for MSS. The length of the longest scan chain is 440 latches. Since all scan latches in MSS and Core are controlled and observed, the test coverage is improved. Also, the test time, which is proportional to the length of the longest scan chain is reduced by 68%.

Core scan chains are divided into 71 pairs, each containing two scan chains. The exclusive-or of the two scan chain outputs of a pair (i) is XORed with the output of MSS scan chain (i) and sent to the MISR bit (i). Since the core and

MSS are completely distinct logically, there is little chance of aliasing.

The 71 bit PRPG is connected to the scan inputs of 142 Core scan chains as follows.To insure that each chain receives a unique pseudo-random pattern, the first 71 scan chains are each fed directly from a separate bit of the PRPG; the remaining scan chains of the Core are fed by the XOR of two distinct PRPG bits. Suppose Core scan chain I is fed by the Exclusive-OR of bits j and k of the PRPG, then no other Core scan chain can be fed by the exclusive-or of the bits j and k of the PRPG.

6.3 LBIST Controller Design and Operation

The function of the LBIST controller is to apply the test patterns generated by the PRPG using the user specified clocking speed and clocking sequence, the number of test patterns and to collect the MISR signature.

The LBIST engine is programmable. The operation of LBIST test is controlled by a set of 32-bit registers. A specific test is chosen by setting these register. The JTAG interface is used to set these registers and read the values of the registers at the end of the test.

An LBIST cycle consists of loading the scan chains, applying the functional clocks and unloading the scan chains.

A typical DC test would have a c1 clock followed by a c2 clock in each cycle. AC tests have two cycles, either starting on c1 clock and ending on c2 clock or starting on c2 clock and ending on c1 clock.

LBIST can test MSS only, Core only and both MSS and Core. The first two modes are for diagnostic purpose.

The PRPG and MISR are for generating pseudorandom test patterns and for compressing the test response similar to the STUMPS architecture. The MISR-enable register is used to mask the output of any scan chain from affecting the MISR. These three registers are 71 bits long.

The TAP controller design has a special command, RUNLBIST that starts the LBIST test. Another special command, SCAN_LBIST_MISR is used to read the final MISR signature.

An LBIST test is started by loading the above described registers and executing the RUNLBIST command. At the end of the test, the final MISR signature is read. The tester then compares the actual MISR signature to the expected MISR signature. The expected MISR signature is obtained by good machine simulation.

For detecting delay faults, at-speed clocking is selected. The clocking sequences are c1-c2-c1-c2-b and c2-c1-c2-c1.

The chain test (no functional clock) is used to check the operation of the PRPG, MISR, MISR-en and the shift clocks.

6.4 Verification of LBIST

The LBIST engine design verification is very important for the success of the test in production. The following key features needed to be checked:

1. There are no X-states propagated to the MISR.
2. The operations of PRPG and MISR are correct.
3. All DC and AC test modes and clocking sequences work correctly.

We used the LBIST simulation capability of an ATPG tool to simulate LBIST to detect if any X- values are propagated to the MISR. Any such design errors detected were fixed.

To verify the operation of PRPG, MISR and to check the various test modes, we used a full core model that allowed us to run LBIST exactly as it would be run on the real core. We simulated each supported LBIST mode on this model and collected cycle by cycle signal values. The signal values allowed us to examine the clocking and scanning of every latch. If the expected clocking and scanning did not occur, we could find the cause of it and fix the error. This tool is extremely valuable because it allows us to look at all internal signals to find and fix errors before the design is released.

7.0 Deterministic at-speed Delay Fault Testing

Like the MPC74xx microprocessors [1,2,3,4,5,6], the e600 core has implemented DFT features that supports at-speed delay fault testing using a tester that runs at much slower speeds than the microprocessor. Here is a summary of e600 design for at-speed delay fault testing:

- The scanning takes place at slow tester speeds, typically 50 MHz. Maximum scanning speed is 100 MHz.
- The core speed can be up to 28 times the scan speed.
- The test is initiated by an asynchronous, non timing critical signal.
- Switching from scan mode to functional mode and from functional to scan mode is at slow tester speed.
- The at-speed high frequency (> 1.5 GHz.) clocks are generated by the clock controller circuit which is part of the core.
- Capability of applying up to 7 functional clock cycles at core-speed.
- A variety of clocking sequences are available for testing. Transition can be launched from either the master latch or the slave latch and the response can be captured at the master latch or the slave latch.
- Test control signals and clocks require no special distribution.

The design supports both broad side transition fault test patterns and path delay test patterns[1,2,3,4,5,6].

7.1 At-speed Clock Generation

A special 5 bit register controls the operation of the at-speed clock controller circuit. Three bits are used to select the number of clock cycles. One bit controls whether the first clock is C1 (master) clock or C2 (slave). The second bit controls whether the last clock is a C1 clock or a C2 clock. These bits are set during scan load of the test pattern. For example, “01000” gives clocking sequence “C1C2C1C2”; “01011” gives clocking sequence “C2C1C2C1”. The at-speed clocking sequence clocks are released by asserting an input pin. As in prior MPC74xx microprocessor designs, the first and last clocks of the clocking sequence are slow clocks. They are not used for at-speed launch or capture. All other clocks are at-speed. The ATPG tool takes this into consideration during test pattern generation. Figure 6 shows the generation of at-speed clocks in the e600 core.

7.2 Test Coverage Enhancement

DFT techniques that were added in e600 to improve the delay fault test coverage are presented in this section.

Analysis of undetected transition faults in MPC7447A showed that the test coverage was adversely affected by sources of X values in AC scan mode. There were two sources of X in AC Scan mode: Non-scan latches and Arrays.

Elimination of non-scan latches:

In MPC7447A, there are three scan chains that can not be scanned and clocked in AC Scan mode.

Another limitation of MPC744A was that a large set of faults could not be detected because some critical latches used in controlling at-speed clocks could not be scanned in AC Scan mode in the full chip model. Thus the full chip model could not be used for generating at-speed transition fault test patterns.

In e600, all scan chains are scannable in all modes. This allows us to generate at-speed test patterns in all models. This required modifications of the clock controls of latches in Zone III. There is one scan chain that is reconfigured in AC scan mode using muxing. This scan chain contains latches that are used for controlling the PLL. In DC scan mode all latches of this scan chain are scannable. In AC Scan mode the PLL control latches are made non scannable.

In e600 we added the capability to freeze Array Clocks. Using this capability, Array outputs are driven to a known value. Thus, Array output is no longer “X” and test patterns can be generated without having to write and read Arrays.

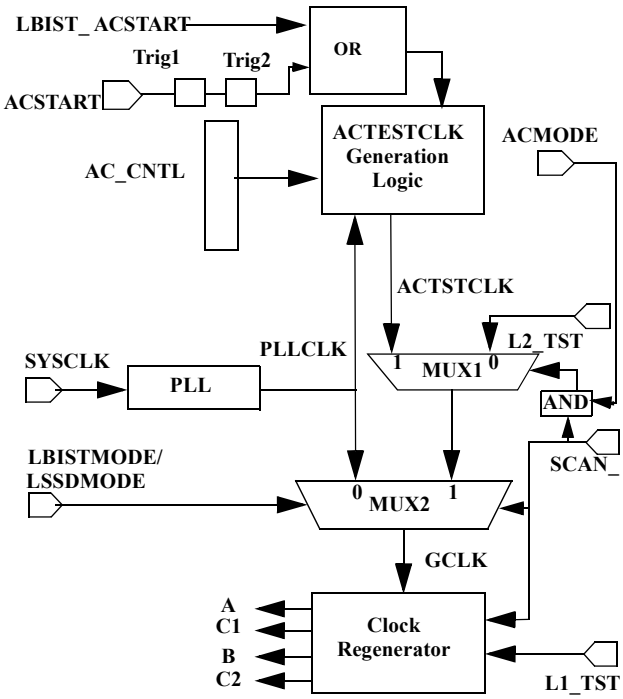

Figure 6: Delay Fault Test Clock Control Circuit

7.3 Increased Delay Fault Detection

Microprocessor’s have a set of paths that do not run at microprocessor speed. This is particularly true of paths that have been modified to increase test coverage by adding DFT circuits. Circuits that are used for TAP, test functions, and debug functions also run at slower speeds. In previous designs, at-speed test covered all circuits, and hence, was run not at the microprocessor speed but at a much slower speed. This resulted in a lower test coverage for delay defects. We recognized that a better design would be to have a test mode, called Fast AC, in which only the circuits running at core speeds are tested. A second mode, called slow AC, is provided to test circuits that run at slower speed. This increases the detection of delay faults.

8.0 Custom Array DFT

The e600 core DFT strategy is to model each custom array, generate test patterns and verify them using switch level verilog simulation. Custom Array designers are given a set a rules that they follow to assure the testability of the arrays. This strategy allows us to detect any problems that may affect test coverage adversely. It also allows us to validate the models for the arrays. These models are used to build a core level model that is used to generate core test patterns. Table 4 below shows the stuck at fault test cover-

age for each array.

Array Name	# of Instances	Size (bits)	Ports (rd/wr)	s-s-a Faults	LSSD Coverage
BTIC	1	~22K	1/1	39228	97.29%
BHT	1	4096	32/2	2440	99.47%
FHIT	1	1024	33/1	8944	97.66%
FPR	2	1120	4/3	1680	90.15%
FRB	2	560	3/2	2116	89.89%
GPR	1	1024	7/3	2334	90.21%
GRB	1	512	9/6	3132	90.71%
VRB	2	1024	9/5	5974	90.23%
VRF	1	2k	7/3	4546	89.99%
MMU	2	2.9k	-	23597	98.0%
DL1cache	8	36k	1/1	43308	85.52%
ITagStat	1	~26k	1/1	21573	98.61%
DStatus	1	3968	1/1	11691	93.16%
DTag	1	~24k	1/1	35750	90.38%
IL1cache	8	~36k	1/1	43224	85.9%
L2Data	64	16k	1/1	3133	72.89%
L2Tag	1	88k	1/1	8514	97.67%
L2Status	1	48k	1/1	3222	99.47%

Table 4. Custom Array Test Coverage

For reference, LSSD DC scan test coverage is also listed for each array. In practice, LSSD patterns that are generated for the BISted arrays, are only used for design debug purposes and for filling ABIST coverage holes (i.e., a port not BISTed in a multi-ported array).

9.0 Discussion and Conclusions

We have presented the DFT enhancements implemented in the e600 PowerPC core. The e600 core evolved from MPC7447A PowerPC microprocessor from Freescale. All proven DFT techniques that allowed MPC7447A to achieve very high test coverage using scan test patterns and Array BIST are also in e600. DFT improvements targeted the at-speed delay fault testing, LBIST, and core testing areas. By eliminating non-scan cells, at-speed delay fault test coverage was significantly improved. We introduced a new at-speed delay fault test mode called slow-ac mode, recognizing that certain circuits cannot be tested at full core speed, but can be tested at a slightly slower speed. This partitioning of circuits based on speed allows testing the majority of circuits at full core speed, which was not previously possible. In LBIST area, e600 has at-speed LBIST capability, which should allow LBIST to detect delay faults. We added the capability to simultaneously test MSS and Core using LBIST, and doubled the number scan chains in Core in LBIST mode to reduce the test time by 68%. The core test interface design of e600 is new to G4 microprocessors family. Core test interface supports at-speed scan testing and DC scan testing in an SOC and debug of core and platform.

We are currently testing MPC7448 which is the first PowerPC microprocessor that has e600. Early results indicate that our strategy of test pattern and design verification using switch level verilog has resulted in extremely high test pattern quality. The test patterns are not only detecting manufacturing defects but are providing information that is crucial to silicon debug. A small set of DC scan test patterns that passed the switch level verilog verification failed on the tester. We found that the ATPG model did not match the design. The root cause was that “TIEX” in the RTL level design file were incorrectly translated to TIE0 due to a program error. This error was corrected by fixing the program and regenerating the affected test patterns. We are currently evaluating the at-speed test patterns. Early results show that they are running at speeds lower than the functional speed. We have identified a set of speed limiting paths using the data logs from the tester. A data log identifies a set of failing latches for each test pattern when the test pattern fails on a tester at a specified speed. It is expected that most of these paths would be test and debug functions. We expect the at-speed test patterns to run at very close to functional speed, once the speed limiting test and debug paths are eliminated by modifying the test patterns. Identifying the speed limiting paths for transition fault test patterns is at present a very time consuming and manual task. This is one area where where ATPG tools can improve.

Acknowledgements

Many individuals from the DFT, Custom Array Design, and Logic design teams contributed to the success of the e600 DFT project. We are thankful to the following individuals for their contributions: Scott Remington, Jing Hao, Hema Ramamurthy, Ken Hicks, Christina Adela, Shayan Zhang, Bob Podnar, Omer Ifthikar, Jason Villanueva, Terence Chung, Aneesha Kolhatkar, Dipal Shah, Parina Vani, Steve Karako, Rajesh Raina and George Joos.

References

[1] Rajesh Raina, Robert Bailey, Dawit Belete, Vikram Khosa, Robert Molyneaux, Javier Prado and Ashu Razdan, “DFT Advances in Motorola’s Next Generation 74xx PowerPC^{TM} Microprocessor,” Proc. of the IEEE Int’l Test Conf., pp.131-140, Sept. 2000.

[2] C. Pyron, M.Alexander, J. Golab, G. Joos, B. Long, R.F. Molyneaux, R. Raina and N. Tendolkar, “DFT Advances in the Motorola’s MPC7400, a PowerPC^{TM} Microprocessor,” Proc. of the IEEE Int’l Test Conf., pp. 137-146, Sept. 1999.

[3] B. Bailey, A. Metayer, B. Svrcek, N. Tendolkar, E. Wolf, E. Fiene, M. Alexander, R. Woltenberg and R. Raina, “Test Methodology for Motorola’s High Performance e500 Core Based On PowerPC Instruction Set Architecture,” Proc. of the IEEE Int’l Test Conf., pp.

574-583, October 7-10, 2002.
[4] D. Belete, A. Razdan, W. Schwarz, R. Raina, C. Hawkins and J. Morehead, “Use of DFT Techniques in Speed Grading a 1GHz+ Microprocessor,” Int’l Test Conference, pp. 1111-1119, October 7-10, 2002.
[5] N. N. Tendolkar, R.F. Molyneaux, C. Pyron and R. Raina, “At-speed scan based testing on MPC7400 Microprocessor,” Proc. IEEE VLSI Test Symposium, pp. 3-8, May 2000.
[6] Nandu Tendolkar, Rajesh Raina, Rick Woltenberg, Xijiang Lin, Bruce Swanson and Greg Aldrich, “Novel Techniques for Achieving High At-Speed Transition Fault Test Coverage for Motorola’s Microprocessors Based on PowerPCTM Instruction Set Architecture,” Proc. 20th IEEE VLSI Test Symposium, April 2002.
[7] Erik Jan Marinissen, Sandeep Goel and Maurice Lousberg, “Wrapper Design for Embedded Core Test,” Proc. of the IEEE Int’l Test Conf.,pp. 911-920,Sept. 2000.
[8] Yervant Zorian, Erik Marinissen and Sujit Dey, “Testing Embedded-Core Based System Chips,” Proc. IEEE Int’l Test Conference, pp. 130-143, September 1998.
[9] IEEE P1500 Web Site. http://grouper.ieee.org/groups/1500.
[10] P. Bardell, W.H. McAnney and J. Savir, *Built-In Test for VLSI: Pseudorandom Techniques,* Wiley Interscience, 1987.
[11] G. Heatherington, T. Fryars, N. Tamarapalli, M. Kassab, A. Hassan and Janusz Rajski, “Logic BIST for Large Industrial Designs: Real Issues and Case Studies,” Proc. IEEE Int’l Test Conference, pp. 358-367, 1999
[12] M. Kusko, B. J. Robbins, T. J. Koprowski and W. V. Huott, “99% AC test coverage using only LBIST on the 1 GHz IBM S/390 z Series 900 Microprocessors,” Proc. IEEE Int’l Test Conference, pp. 586-592, 2001.