

A Test Generation Methodology for High-Performance Computer Chips and Modules

by

Bernd Könemann
c/o IBM Corp.
B56/901-3
Box 390
Poughkeepsie, NY 12602

Phil Noto
c/o IBM Corp.
3A1/306
Route 52 East Fishkill Facility
Hopewell Junction, NY 12533

Abstract

Modern high-performance computer systems use fast complex chips and dense packaging on multi-chip modules to attain top performance characteristics. The complexity and cost of such packages demand very effective and efficient test generation methods for the chips and modules.

A test methodology using LSSD based test generation for the chips and STUMPS based Self-Test for the modules has been developed. Special emphasis is placed on exploiting the on-product clock generation logic for delay test purposes, and on an efficient method for initializing embedded memory elements. The resultant methodology takes not only test generation requirements into account, but also the particular needs of debug and failure analysis.

1. Introduction

The power of modern computer systems is continuously enhanced by increasing the density and speed of chip and packaging technologies. Advanced Multi-Layer substrate modules (similar to the Thermal Conduction Modules, TCMs, known from commercial mainframe computer systems [1]) are capable of holding well over 100 high-performance bipolar chips. Such tight packaging not only increases the density of logic function, but also contributes to a steady reduction in operating cycle times.

Cycle time reduction places an ever higher demand on the accuracy of the system clocking architecture. Small clock skew is best accomplished by on-module and on-chip clock generation and control logic, which derives local clock phases from a common oscillator signal edge. One of the challenges of testing the resulting high-performance logic modules is to include the clock generation logic into the test in a practical and meaningful way. The basic LSSD [2] test approach sidesteps the issue of clock-generation testing by insisting on multiple level-active test clock signals that can be controlled from separate inputs to avoid race-conditions

during testing. Unfortunately, it is no longer possible to provide an accurate enough delay test of chips and modules through the artificial and relatively inaccurate LSSD test clock distributions. Thus, it has become necessary to develop enhanced design and test generation procedures that allow for incorporating the functional clock generation logic into an LSSD based methodology.

Deeply embedded multi-port memory arrays are becoming more and more frequent design elements in high-performance systems. The embedded memories require special treatment for both, testing the memories proper and testing the complex logic surrounding them. An accurate access time test is one of the central requirements for memory testing. LSSD provides for a structured method of accessing embedded memories such that pre-defined test patterns can be applied regardless of the complexity of the surrounding logic [3]. LSSD, however, demands multiple level-active clocks to separately control the change of address/data registers and the memory access (e.g., the memory write clock signal). The access time test under these conditions is not only limited by the skew in the separate on-chip clock distributions but also by the pin-to-pin skew of the test equipment. To avoid the pin-to-pin skew penalty for functional access, the chips contain special clock generation circuits, which derive the address/data register and memory access clock phases from a common clock input. The LSSD test methodology needed to be enhanced to allow for exploiting these clock generation facilities for testing the embedded memory arrays as well the surrounding logic.

The STUMPS [4] Self-Test method used for testing the modules employs signature generation to compact the test response data. Signature based testing requires that no unpredictable states propagate to the signature generation logic and render final signature unpredictable. Consequently, all embedded storage elements, including the embedded memories, whose outputs could propagate to a signature generator have to be initialized to a known starting state. An appropriate method for memory initialization requiring little or no extra logic in the

product under test, and minimal storage/hardware in the test equipment had to be developed.

Although chip level delay testing is effective in weeding out many delay related defects, this alone is not sufficient. New defects can be introduced during module assembly, in particular defects affecting the chip periphery and the chip-to-chip paths. Hence, module level delay test including the chip-to-chip paths is needed to improve the module quality levels, and a module delay test capable of utilizing the on-module clock generation circuits had to be incorporated into the STUMPS methodology.

The separate test clocks required for LSSD operation have historically proven to be invaluable for diagnosis and debug in that they allow to establish a race-free operation by slowing down the test clocks. This makes it possible to "bypass" short path failures (caused by marginal design, defects, or test equipment limitations) and thereby distinguish between static and dynamic problems. To retain such diagnoseability and debug advantages, the test methodology ensures that all major tests can be performed using the on-product clock generation circuits or the LSSD test clocks interchangeably.

Finally, the state-of-the-art module technology provides for well over 1500 signal I/O pins (in addition to a large number of power supply pins). A fully configured tester with high-performance digital tester channels for all signal I/O pins tends to be extremely expensive. Hence, a reduced pin count testing strategy (similar to the one described in [5] for chip testing) was adopted for the module test step. The module tester has fully configured channels only for a fixed, small subset of the signal I/O pins (around 100). The remaining module I/O pins are contacted in separate DC parametric test step. The module design and test methodology takes the reduced pin count testing constraints into account.

2. Clocking System Overview

The structure of the product clocking system is of great importance for the test generation methodology.

The main functional clock input to the module, as shown in Figure 1, is the signal called "Osc". This incoming oscillator signal is split into two major clock phases, C1 and C2, before feeding the logic chips on the module. Each chip contains dedicated Clock Generation & Control circuitry, which in normal functional mode shapes and propagates the C1 and C2 phases to the L1 and L2 latches on the chip. If the chip contains embedded memory arrays, a third phase called WrtC is generated from the C2 phase to control the memory write operations.

This basic functional clocking scheme is not com-

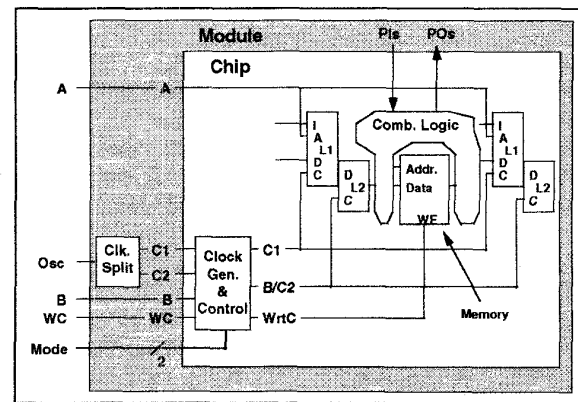


Figure 1: High Level View of Module/Chip Clock Generation Hierarchy

patible with the fundamental LSSD clocking constraints. LSSD requires that the constituent storage element types of the design (that is the L1s, L2s, and memories) are controllable from **separate** test clock inputs. The Clock Generation & Control circuitry provides the ability to control the derived functional clock phases from dedicated clock Primary Inputs (PIs) as needed for test and debug. The module level LSSD test clock signals are A, B, and WC in addition to the oscillator (Osc) input.

Two mode select signals, "Mode", control the switching between functional mode and the DC and AC test modes. In DC test mode at the module level, the C1 phase for the L1s is controlled from the Osc input, the C2 phase for the L2s from the B input, and the WrtC clock phase for the embedded memories from the "WC" input. The application of each phase can, thus, be controlled and timed individually to prevent internal race conditions from affecting the test responses. In functional operation and in AC test mode, all internal clock phases are derived from the common Osc input. At the chip level, the C1 and C2 clock phases are available as separate PIs.

Figure 1 also illustrates how the basic storage element types and associated clock signals are used on the logic chips. Registers are implemented as Master/Slave latch pairs (L1 Master latch controlled by the C1 clock phase, and L2 Slave latch controlled by the C2 phase). For scanning, the L1s are equipped with a second port for the scan data (L1 input "I") which is controlled by the shift "A" clock, and a designated shift B clock controls the transfer of scan data from the L1s into the L2s. Embedded memories are sandwiched between L2s or chip inputs ("PIs") on the input side (for Address and Data In) and L1s or chip outputs ("POs") on the output side. All memories are so called "hot read" memories in that the output data propagate the contents of the memory word pointed to by the read address without buffering.

Writing, on the other hand, is controlled by the WrtC signal which feeds the Write Enables (WE) of the memories. For clarity, Figure 1 does not show any additional clock gating that may exist for functional reasons.

3. Chip Level Test Generation

Since the chip clocking interface shown in Figure 1 provides for an LSSD-compatible mode of operation, it is possible to use conventional LSSD test generation for the chips. The chip release criteria call for a DC stuck-at fault coverage over 99.5% for each chip.

3.1 Chip Level Logic Test Generation Method

For logic test generation (that is for testing the logic surrounding any embedded memories), no major changes to the standard methodology are needed and the LSSD test generation system is generally capable of meeting or exceeding the test coverage objectives.

The resulting DC stuck-at fault tests are applied at DC speeds (static) as well as under more realistic AC timing conditions as described in some detail in [6]. The timing data for the AC test application are derived by chip level timing analysis and submitted together with the DC chip test data. The conversion of the DC test data into the AC test format is done in the chip manufacturing house and is transparent to the chip designers.

3.2 Chip Level Embedded Memory Test Generation Method

Besides an effective test for the logic, it is necessary to also provide comprehensive tests for the embedded memories. In the LSSD test system, memory tests are not automatically generated from a stuck-at fault model. Instead, the memory designers, based on their intimate knowledge of the memory circuits and layout, compile appropriate test pattern sets for each memory macro type offered in the design library. These macro test patterns define test stimulus values at the macro inputs and expected response values at the macro outputs. Unfortunately, the inputs and outputs of embedded macros are not directly accessible from the tester. Thus, a method for mapping the macro test data from the macro boundary to tester accessible points is needed for applying the tests. The LSSD test system uses a concept of macro isolation by "1:1 Correspondence" for that purpose. For the basic ideas behind "1:1 Correspondence", the readers are referred to reference [4].

For the purposes of this discussion it is sufficient to know that the logic network is pre-conditioned such that each macro input needed for particular memory operation is directly controllable from a unique PI or LSSD latch (e.g., L2s) and each macro output involved in the

operation is directly observable at a PO or LSSD latch (e.g., L1). Once this correspondence is established, the macro test data are easily translated into PI stimuli, LSSD shift register loads/unloads, and PO measurements. Figure 2 illustrates the concept with a simple example. In the example, 1:1 correspondence has been established between a Latch, L2_234 and a macro input, In_1. L2_234 is directly controllable from the tester via an LSSD scan operation. The pre-conditioning value, a logic '1', on the chip input is required to sensitize the AND gate such that value loaded into L2_234 is propagated to In_1. Thus, the tester implicitly has direct access to the input of the embedded macro.

The test generation software system contains a tool that automatically finds appropriate correspondence points and determines the required pre-conditioning for any specified (sub-)set of macro inputs/outputs. If the surrounding logic does not support such correspondence, the chip designers have to modify the logic for correspondence. This, however, does not happen very frequently.

By manually specifying preferred correspondence points and/or setting appropriate pre-conditioning values, the correspondence tool can be directed towards specific correspondence configurations, if so desired. This feature

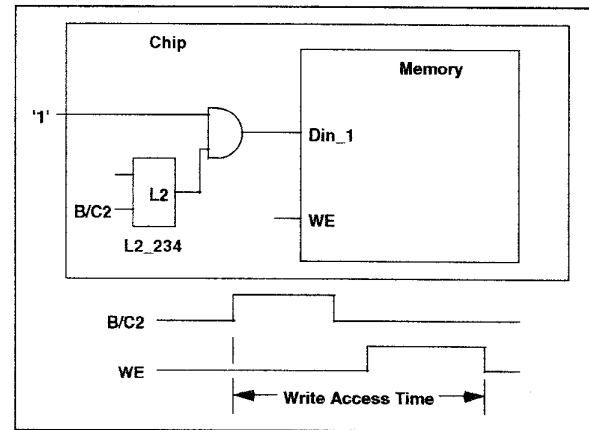


Figure 2: Simple Pre-Conditioning/Correspondence Example

is utilized here to control the correspondence to the embedded memories for purposes of AC testing the memory write operation. The normal LSSD correspondence for the Write Enables of the embedded memories uses the LSSD Write Clock pin, WC. To establish this correspondence, the Mode pins (see Figure 1) are set to indicate DC test mode. The Mode pins are dedicated pins on all chips with embedded memories. Hence, it is easy to generate the respective

correspondence control statements for these pins automatically when generating the chip net lists.

The LSSD correspondence is not suitable for a very accurate access time measurements of the embedded memory, because the distribution for the LSSD Write Clock, WC, is not designed with very tight tolerance in mind. Combined with the inherent pin-to-pin skew of the test equipment, this limits the attainable measurement accuracy.

To provide for a more accurate access time test, a second type of correspondence is generated. This AC correspondence is established by setting the Mode pins to AC test operation in which the Write Enable signal is derived from the C2 clock. The memory inputs are fed from L2s (see Figure 1) under the control of the C2 clock. When the C2 clock is pulsed, the address and data input values change. The derived Write Enable is "automatically" generated from the same clock pulse. Thus, a tightly timed write access is internally produced from a single clock pulse and the pin-to-pin skew on the tester can not degrade the measurement.

A very important aspect of the methodology is that both types of correspondence can be used with the same test patterns. This is crucial for diagnosis and debug. With the fixed on-chip timing offset between the C2 clock and the Write Enables, for example, it is not possible to distinguish between DC and AC defects. This distinction can, however, be made by switching to LSSD mode correspondence and repeating the same test(s) under various timings controlled from the tester. Practical experience from the past has shown that this not only helps to identify DC problems, but often helps to characterize gross AC defects (that is defects large enough to show up despite guardbanding the tester timing for pin-to-pin skew). From an engineering point of view, the DC single step operation possible in LSSD mode has historically proven to be invaluable for debugging parts with marginal timing design.

3.3 Chip Test Generation Flow

Figure 3 shows the major steps and components in the chip test generation process flow and how it is integrated with the physical design process to assure data integrity. The major steps and components shown in Figure 3 are:

- Design Data Base

The Design Data Base is the common repository for all physical and test generation model data that are needed for chip manufacturing. Built-In Audit and Checking facilities keep track of which levels of source net list data, cell library data, and technology rules were used to generate the physical and test data. This

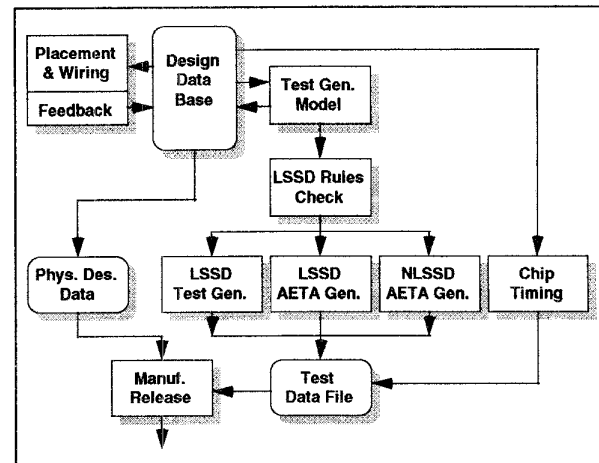


Figure 3: Chip Test Generation Data Flow

information is used to assure that the physical design data and the test data released to manufacturing are consistent.

- Placement & Wiring

In this physical design process, the masterslice cells are placed and wired on the chip underlayers. The placement and wiring results are fed back into the Design Data Base, where they are checked and registered together with their Audit Data.

- Test Generation Model

For performance reasons, the test generation software system builds its own working model from the data in the Design Data Base.

- LSSD Rules Check

This step checks the network under test for LSSD rules compliance [2, 4]. The results of this check are audited in manufacturing release.

- LSSD Test Generation

The LSSD test generation system is based on a static single stuck-at fault objective. The system relies on compliance with the LSSD rules to insure that the generated test data are race-free. A DC stuck-at fault coverage of well above 99% is required and typically achieved for each chip.

- LSSD AETA Generation

This procedure generates the data for DC testing of the embedded memory arrays. The correspondence data (see section 3.2) and memory test patterns are combined into a so called All Events Trace for Arrays, AETA, file.

- NLSSD AETA Generation

This procedure generates the alternate test data for the embedded memory arrays, which allow for internally timed AC conditions for writing (see section 3.2). The correspondence data and memory test patterns are written into an alternate All Events Trace for Arrays, AETA'.

- Chip Timing

The chip timing data are created by a Block-Oriented Timing Analysis program based on gate and wire delays in the chip after final placement and wiring data have been fed back into the Design Data Base. Timings are provided for various path delay types involving Primary Inputs, Primary Outputs, Latches, and memory arrays. The timing data are used in converting the DC LSSD tests into delay tests (see [6] for more details). The conversion is done after manufacturing release.

- Test Data File

The Test Data File accumulates results from the various test generation steps and from chip timing. Also, a complete description of the test generation model and cross-references to the physical design data are added. The Test Data File is not fed back into the common Design Data Base, but enough Audit information is included to uniquely identify the source net list, library, and rules data used.

- Manufacturing Release

Manufacturing Release combines the chip physical design data and the test data into a complete data package for chip manufacturing. The Audit information from physical design and test generation is compared to make sure that the physical design data and the test data are indeed consistent.

4. Module Level Test Generation

The module level test generation methodology is also based on the LSSD philosophy. The Clock Generation & Control circuitry implements the necessary interfaces for LSSD mode operation. A key difference between chip level and module level test generation is the use of Self-Test at the module level, whereas a fault-oriented stored pattern test method is used at the chip level. The module level Self-Test has an objective of exceeding 95% stuck-at fault coverage.

4.1 Module Level Logic Test Generation

Each module contains a built-in Pseudo Random Pattern Generator (PRPG) and Multiple Input Signature Register (MISR) in a STUMPS configuration. The STUMPS configuration is well described in reference [3]. The PRPG and MISR communicate with the functional logic through LSSD scan strings in the functional logic.

The scan strings in the STUMPS environment are called STUMPS Channels.

The module tester has -for cost reasons among other things- fully configured tester channels only for a fixed subset of pins (roughly 100 out of more than 1500). The remaining pins are contacted in a separate test step for DC parametric measurements. Hence, the vast majority of the module pins cannot be controlled or observed for digital testing. Most module I/Os happen to be functionally latched and it is "natural" to use the associated latches in a Boundary Scan arrangements (see reference [5] for an overview of Boundary Scan in relationship to Reduced Pin Count Testing). The uncontrollable inputs are degated or made to float to a fixed logic value in test mode (DC and AC). This avoids unpredictable states from entering the module during Self-Test. The logic model for test generation reflects the degated or constant input states. Likewise, the model reflects the inability to take test response measures at the majority of module outputs. The Boundary Scan latches themselves are part of the product STUMPS channels and can, thus, be loaded from the built-in PRPG and unloaded into the built-in MISR. Consequently, almost all logic except some receiver and driver logic around the module periphery is digitally testable through the Self-Test facilities. The connectivity and parametric integrity of the peripheral circuits are tested in the separate DC parametric I/O test.

One central objective of the logic Self-Test methodology is to support tightly timed AC tests using the on-module Clock Generation & Control circuitry. For diagnostic and debug reasons, as discussed earlier, it is considered crucial that the same tests can be executed under the tightly timed AC conditions as well as under DC conditions. The Clock Generation & Control logic is designed with that objective in mind.

In normal functional mode, each Osc pulse internally generates a sequence of C1, C2, WrtC clock pulses. To simplify failure diagnosis, it was requested to suppress the leading C1 phase for testing purposes (the leading C1 phase replaces scanned-in values in the L1s with data from a "previous" time image, which then would have to be included in the diagnostic process). Suppressing the initial C1 phase does not affect the basic delay test. The leading edge of the C2 phase is used to release transitions at the outputs of L2s. These transitions propagate through the logic and are subsequently captured into the receiving L1s by the trailing edge of the C1 phase. A delay defect is detected, if the propagation of the transitions is delayed sufficiently to latch up the wrong value in one or more of the receiving L1s.

Before scanning the captured responses out to the MISR, they must be copied from the L1s into the L2s. This is done by pulsing either the C2 or the B clock. In

AC test mode the C2 is used, such that the sequence of events is C2->C1->C2. This is followed by a Scan operation to unload the responses into the MISR and to load new pseudo-random test stimuli for the next test from the PRPG. The normal functional operation of the Clock Generation & Control logic does not produce the desired sequence. Hence, a small Finite State Machine has been included in the Clock Generation & Control logic, which in AC test mode inhibits the leading C1 clock that would normally be generated in the first Osc cycle, but includes it in any subsequent cycle. The WrtC clock pulse can be suppressed, if desired, by appropriate gating conditions.

Figure 4 illustrates how the internal C2->C1->C2 sequence can be generated by two consecutive Osc pulses in AC test mode. The consecutive Osc pulses are applied under functional timing (that is with functional cycle time between the pulses). Thus, the timing offsets between the derived phases matches the offsets under functional operating conditions. Since only a single input, namely the Osc input, is involved, the tester pin-to-pin skew cannot degrade the test. The Osc input is assigned

Note: <WrtC> indicates that WrtC can be suppressed		
	1. Osc. Pulse	2. Osc. Pulse
Normal Funct. Mode	C1->C2->WrtC	C1->C2->WrtC
AC Test Mode	C2-><WrtC>	C1->C2-><WrtC>
Corr. DC Clocks	B-><WrtC>	C1->B-><WrtC>

Figure 4: Internally Generated Clock Wave Forms for AC Test Mode

to the same fixed physical pin location on all modules and the associated tester channel is equipped with a precision oscillator to further improve the timing accuracy of the test.

An important by-product of the C2->C1->C2 sequence is the ability to detect not only long path problems (propagation delay defects on paths from the L2s to the L1s) but also short path problems. The short path problems are uncovered by the last two of the three clock pulses. The C1->C2 clock pulses derived from the second Osc pulse are slightly overlapped. This can cause race-through problems if transitions released by the last C2 clock reach the L1s before the hold time on the C1 clock has expired. One particular defect that can cause this symptom is an excessive delay in a C1 clock driver turning off.

The clock driver circuits are designed such that in DC test mode the C2 clock can be logically replaced by

the B clock, and the WrtC clock can be logically replaced by the WC clock. This clock substitution can be done without changing the expected test response values. The DC sequence corresponding to the C2->C1->C2 AC test sequence is B->C1->B. To conserve computer resources, only the DC sequence is explicitly simulated using the IBM Self-Test support software system [7].

The AC test sequences are derived by post-processing the DC sequence data from the Test Data File. Because of the careful implementation of the Clock Generation & Control logic, all results (that is the simulated good-machine signatures) should match between DC and AC test modes and no separate calculation of the AC test signatures is needed. The ability to run the same tests under both, AC and DC, conditions is very important for distinguishing between AC and DC defects in failure diagnostics.

4.2 Embedded Memory Initialization

Embedded memories are not scannable like the other storage elements (that is the L1s and L2s) and are, therefore, not loaded by the normal Scan operations. Since the memories contents are unpredictable after power-on, it is necessary to provide an alternate method for initializing the memory cells to known values prior to Self-Test.

To minimize the design impact for supporting Self-Test, it was decided not to use built-in address steppers for memory initialization as, for example, described in reference [8]. Instead, the initialization data are completely supplied from the tester utilizing memory correspondence and data mapping concepts similar to those discussed in section 3.2. A dedicated memory initialization mode is provided, in which the module scan chains are not fed from the built-in PRPG, but from the tester. The objective is to initialize the memories with predictable pseudo-random values.

To control the memory write operations from the tester, simultaneous correspondence between L2 latches and memory write address and data inputs is established simultaneously for as many memories as possible (memory grouping). Data inputs, write enables, and write address inputs of the memories are specially marked in the memory macro models. The write address inputs of the memories are further "labelled" according to bit order in the address (from Least Significant Bit to Most Significant Bit). A scan chain map is generated in which each latch corresponding to a memory macro input or needed for pre-conditioning is labelled accordingly.

Figure 5 illustrates the labelling for a simple example. Within each scan chain, any SRL contributing to the pre-conditioning is labelled with the required pre-conditioning value (e.g., '1' in the example). The write

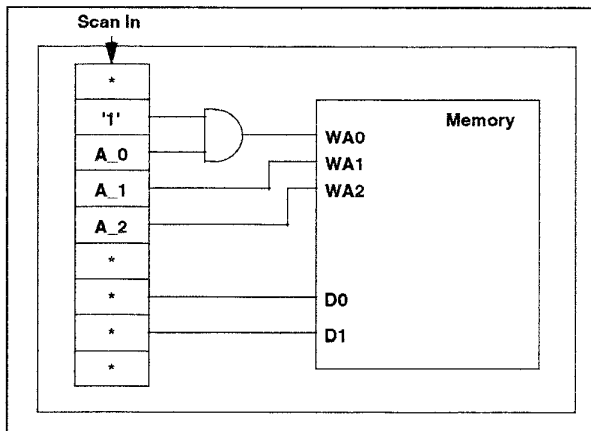


Figure 5: Scan Chain Labelling Example

address correspondence SRLs are labelled according to the bit order found in the macro model (e.g., A_0 indicates that the SRL corresponds to the least significant bit, bit 0, of a memory write address). All other SRLs, including the data input correspondence SRLs, are labelled as "other" (indicated by the symbol * in Figure 5). Based on this information a scan chain map is created for each module scan chain. The map identifies the type of each SRL in the associated chain.

During memory initialization, the scan chains are loaded with different types of logic values depending on the scan chain maps. Pre-conditioning SRLs get loaded with the respective values found in the map, write address correspondence SRLs get loaded with the respective bit from a global address counter, and all other SRLs get loaded with pseudo-random values. When the loading is completed, the memory write clock is pulsed to write pseudo-random values into the words selected by the write address correspondence SRLs. The global address is incremented between loads until the address space of the deepest memory in the currently selected initialization group has been covered. The process must be repeated for any other groups until all memories on the module are initialized.

The memory initialization can be executed statically or at AC speed. In the static version, the WC clock input is used for writing the memories. For AC test, the last B clock of the scan chain load and the WC clock are replaced by a single oscillator pulse in AC test mode. This will internally generate tightly timed C2 and WrtC clock phases to stress test the memories for write access time defects.

During the logic testing, the memories are completely initialized 32 times, each time using the same addressing sequence and pre-conditioning, but different pseudo-random values. This is known to be sufficient for

exhibiting practically all cell stuck-at and write address decoder faults in the memories ([9]). The read paths and read decoders are tested implicitly by reading the initialized memories during the logic test.

4.3 Module Test Generation Flow

The module level test generation flow is similar to the chip level test generation flow, and is shown in Figure 6.

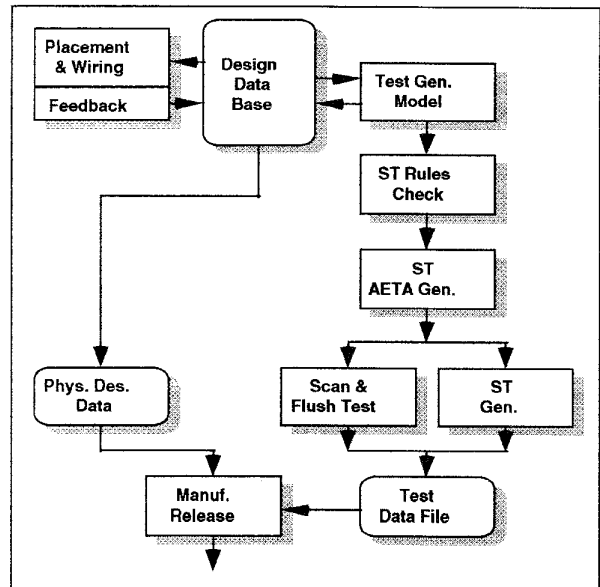


Figure 6: Module Test Generation Flow

- Design Data Base:

The Design Data Base is the common repository for all physical and test generation model data that are needed for module manufacturing. Built-In Audit Trace and Checking facilities keep track of which levels of source net list data, chip design data, and technology rules were used to generate the physical and test data. This assures that the physical design data and the test data released to manufacturing are consistent.

- Placement & Wiring:

This is the physical design process in which the chips are placed and wired on the substrate. The placement and wiring results are fed back into the Design Data Base, where they are checked and registered together with their Audit Trace.

- Test Generation Model:

The model generated for Self-Test contains some special macro blocks representing the on-product Pseudo-

Random Pattern Generator (PRPG) and Multiple-Input Signature Register (MISR). Also, the characteristics of the Pattern Generators and Signature registers of the module tester are included.

- **ST Rules Check:**

This step checks the network under test for compliance with the LSSD and Self-Test rules. The results of this check are audited in manufacturing release.

- **ST AETA Generation:**

To initialize the embedded memories, these are isolated as described in section 4.2. The isolation data are captured into an All Events Trace for Arrays (ST AETA) for Self-Test initialization.

- **Scan and Flush Test:**

Scan and Flush tests are generated to validate the integrity of the on-product LSSD scan chains. Also, a Channel Scan test is produced to verify the operation of the Scan Channels in the STUMPS configuration.

- **Self-Test Generation:**

This procedure generates the pattern generator seeds and calculates the resulting signatures for the Self-Test. Prior to simulating the logic test, the 32 different array initializations are simulated using the isolation data in the ST AETA.

- **Test Data File:**

The Test Data File accumulates results from the test generation steps. Also, a complete description of the test generation model and cross-references linking chip locations, pins, and substrate nets with the internal indices used in the Test Generation Model are added. The Test Data File is not fed back into the common Design Data Base, but contains enough Audit information to uniquely identify the source net list, chip design levels, libraries, and rules data used.

- **Manufacturing Release:**

Manufacturing Release combines the module physical design data and the test data into a complete data package for module manufacturing. The Audit Trace information is used to make sure that the physical and the test data are indeed consistent.

5. Results

To illustrate the scope of the described test generation system, Figure 7 shows a brief summary of statistics from a sample of 4 modules.

The test coverage includes only stuck-at faults. Faults outside of the Boundary Scan, which are not

	Blocks	Faults	Arrays	Tests	Coverage
1	1.9M	2.1M	1013	134K	99.2%
2	0.9M	1.3M	686	134K	97.3%
3	0.2M	0.3M	0	134K	96.9%
4	2.5M	3.8M	256	134K	97.1K

Figure 7: Summary of Test Generation Results

reachable in Self-Test mode, have not been subtracted from the fault list. Thus, the test coverage in terms of faults testable in Self-Test mode would be slightly higher.

Acknowledgements

Any large scale project as the one described in this paper, involves many people from different sites and disciplines. Our special thanks go to Paul Bardell, Bill McAnney and Marc Shulman who helped to set the direction for the module test methodology, to all the chip and module designers for contributing to and following the design for test guidelines, and the test generation software developers for their continuous and expeditious support.

References

- [1] A.Blodgett and D.Barbour, "Thermal Conduction Module: A High Performance Multilayer Ceramic Package", IBM Journal Research & Development, Vol. 26, pp. 30-36, 1982.
- [2] E.Eichelberger and T.Williams, "A Logic Design Structure for LSI Testability", Proc. Design Automation Conference, pp. 462-468, 1977
- [3] P.Bardell and W.McAnney, "Self-Testing Multi-Chip Modules", Proc. International Test Conference, pp.200-204, 1982.
- [4] E.Eichelberger et al., "A Logic Design Structure for Testing Internal Arrays", Proc. 3rd USA-Japan Computer Conf., pp. 266-272, 1978.
- [5] R.W.Bassett et al., "Low-Cost Testing of High-Density Logic Components", Design & Test, pp. 15-28, April 1990.
- [6] F.Motika et al., "A Logic Chip Delay-Test Method Based on System Timing", IBM Journal Research & Development, Vol. 34 (No. 2/3), pp. 299-313, March 1990.
- [7] B.L.Keller and T.J.Snethen, "Built-In Self-Test Support in the IBM Engineering Design System", IBM Journal Research & Development, Vol. 34 (No. 2/3), pp. 406-415, March 1990.
- [8] C.W.Starke, "Design for Testability and Diagnosis in a VLSI CMOS System/370 Processor", IBM Journal Research & Development, Vol. 34 (No. 2/3), pp. 355-362, March 1990.
- [9] W.H.McAnney et al., "Random Testing for Stuck-At Storage Cells in an Embedded Memory", Proc. International Test Conf., pp. 157-166, 1984.