

# A 370-MHz Memory Built-In Self-Test State Machine

R. Dean Adams, John Connor, Garrett S. Koch, Luigi Ternullo Jr.  
IBM Microelectronics Division  
Essex Junction, Vermont, USA

## Abstract

Hardware and simulation results for a 370-MHz memory built-in self-test state machine are presented. Dynamic differential cascode voltage switch logic, unique clocking techniques, and logic pipelining were used to achieve the 370-MHz performance. Testing of multiple SRAMs and content addressable memories is accomplished with deterministic patterns generated by the state machine. Inclusion of a programmable pattern implemented via scan initialization provides test-pattern flexibility. Failing addresses are stored for redundancy implementation.

## Introduction

Static memory speeds continue to increase, forcing test circuitry to accelerate as well. High-speed embedded memories require built-in self-test (BIST) circuitry to thoroughly test the memory at speed[1]. Although high-speed self-test circuits for memories exist, most have been dependent upon pseudorandom patterns [2]. This approach can be helpful in finding non-target-type defects[3] but, for well-understood defect sets, deterministic test patterns are superior. The exact patterns needed to exercise and find the defects can be further simplified because the memory configurations are understood. When considering static CMOS memories only, the potential defect set is further narrowed. A deterministic pattern can be quite short whereas a pseudorandom pattern may require many vector combinations to detect a known defect mode.

This paper describes a hardware-verified 370-MHz memory BIST state machine. The BIST chip was fabricated using a 2.5-V, 0.25- $\mu\text{m}$ -channel CMOS process. The test site was functional with two levels of metal plus a M0 level. After completing characterization of the BIST only chip, the state machine was further designed and fabricated with a group of several memories.

## Circuits and Clocking

Designing a deterministic state machine for 370 MHz and faster required careful choice of a design methodol-

ogy. After reviewing various options, a domino differential cascode voltage switch (DCVS) technique was selected[4]. The DCVS technique is faster than standard static CMOS logic but maintains thoroughness of logic testability[5]. Test circuits should be inherently more testable than surrounding circuits. Although Chu found a 55% performance improvement by using DCVS over static CMOS logic, we found that by doing a full custom static design the difference was only 20%.

A single external clock was used to generate three clocks: the reset and data launch clocks, which were chopped to 0.5-ns-wide pulses; and the data capture clock, which was a function of the cycle time (Fig. 1). All operations are triggered off the falling edge of the system clock, the only requirement being that this clock be more than 0.6-ns wide. The capture clock, which has been active since the latter part of the previous cycle, turns off immediately after the fall of the system clock. A 0.5-ns wide reset clock, which is active low, then fires, and resets the dynamic logic and dynamic latches. A feature included in the clock generator allows the reset clock to be extended by 1.0 ns for diagnostic purposes; ie. if a characterization problem occurs, the reset clock can be extended to ensure sufficient reset time duration. As the reset clock becomes inactive, the launch clock pulses, activating the dynamic latches. Subsequently, the leading edge of the capture clock occurs.

The logic design chosen accentuated performance, not area, which resulted in a full-scan design with dynamic slave latches feeding the DCVS logic. A separate static slave latch served as the scan portion to further optimize the latches for performance. The DCVS was exclusively precharge low drive-high logic. Because of the complexity and area overhead, DCVS logic was used only when performance necessitated. This allowed the BIST state machine to occupy only 2% of the area whereas previous state machines could require 5%[6].

Modified-ratio drive stages are used so that the evaluation phase of the logic can perform faster than the reset portion. The dynamic logic uses a slight skew of 5:2 PFET to NFET width. A larger skew actually degrades cycle performance because a larger portion of the cycle and a wider reset clock are needed to reset the circuits. Eight pipeline stages were used from initial address launch to failing address capture, to achieve 370-MHz

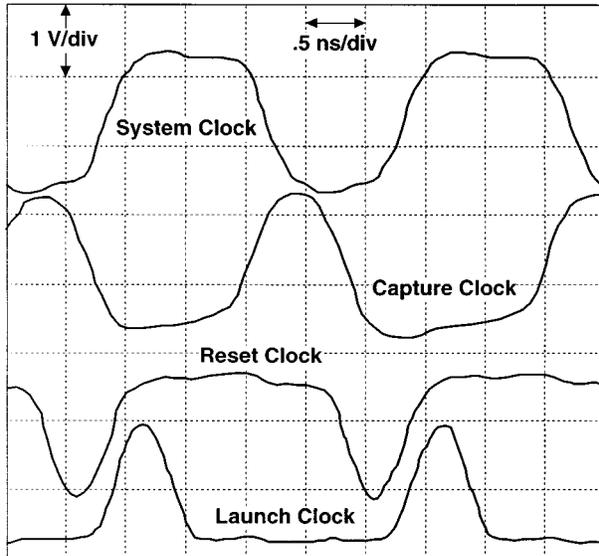


Figure 1. BIST clocks.

operation. Figure 2 shows the highlights of each pipe stage calculation.

### Data-Generation Techniques

The BIST state machine supplies address, data, and control signals to a memory under test (Fig. 3) and can apply six patterns to a memory: programmable (PG), unique address ripple word, unique address ripple bit, word-line stripe, checkerboard multiple read, and blanket initialization test[7]. Each test pattern contains two or three subcycles which define the number of times a cell is accessed and the read/write (R/W) sequence. The programmable pattern is specifically designed with maximum flexibility to implement patterns not originally anticipated when the memory was designed. During this

Pipe Stage	Function
1	Pre-Address
2	Data Generation
3	Rebuffering
4	Inputs Sent to Memory
5	Memory Exercised
6	Data Outputs Compressed
7	Result Sent to Central Compression
8	Failing Address Captured

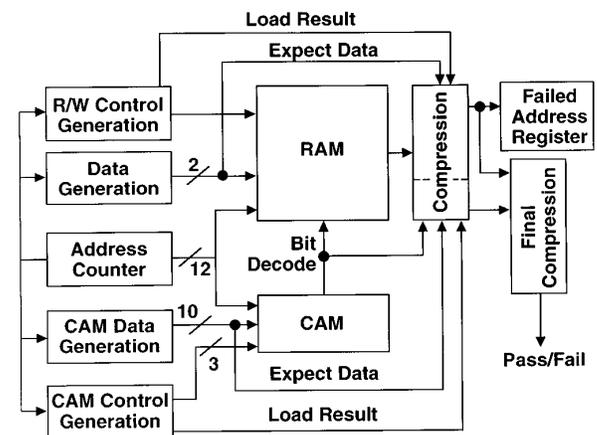
Figure 2. Logic pipe stages.

pattern, data and R/W sequencing are obtained from the eight programmable-data and eight programmable-read/write control latches initialized during scan-in. Programmable options include inverting data between subcycles, inverting data on the word- or bit-address least-significant bits, complementing the odd data bit, and the ability to program eight-, four-, or one-bit data and R/W sequences. Each set of eight latches is wired to form an eight bit shift register with one output. If there are eight operations performed to an array cell at a given address, the eight bit shift register rotates through its entire contents, generating a new output in each cycle. If four operations are performed to a cell, the shift register rotates through half its contents. Shifting is disabled if one operation is performed with data and R/W control supplied from the latch during the shift register's output. Using a broadside load/shift register approach eliminates timing concerns associated with calculating data and R/W control for each cycle. In subsequent patterns, data and R/W control are broadside-loaded from combinatorial logic in the state machine during single-cycle WAIT states, which occur when the address limit is reached. The last pattern leaves the memory in a blanket-0 parity-clean condition.

### Data Compression/Failed Address Register

Expect data is generated by the state machine and compared with memory output. Even and odd data bits exit the memory and are captured by latches that generate precharged low, true and complement output pairs. These output pairs are compressed by DCVS OR gates and then compared to their respective even and odd expected data. A 24-bit OR function, performed across a 2000-nm-wide memory, compresses exiting data; the compression is completed in only 0.74 ns for a single-bit mismatch. The compression is gated with a Load Result signal, which allows selective compression of data only for those memories undergoing valid tests.

For memories with redundancy, the address of a fail is stored in a failed-address register (FAR)[8]; the FAR,



implemented with the state machine, can store two redundant word addresses. The failed-address function is performed on a subarray basis to increase the yield through more flexible redundancy implementation. The first fail discontinues the loading of any other addresses in the first register; all other fails that occur at the stored word address are ignored. When another address fails, it is stored in the second register. If an additional address fails after all the redundancy is utilized, an overflow bit is set. The scanned-out failing-address data provides a one-to-one redundant address fuse correlation. In addition, a cumulative fail bit of all memory fail signals can be sent off-chip to generate a cycle-by-cycle address fail map.

### CAM Built-In Self-Test (CAMBIST)

Content addressable memories (CAMs) are frequently included with conventional SRAMs in embedded memory macros. Testing CAMs with BIST requires testing both the memory and the logic of the CAM with deterministic patterns. The memory is tested with the standard memory patterns already described while the logic is exercised by stuck fault type patterns. All these patterns are generated by the state machine. The memory combination being tested used a CAM to perform associative decode bit addressing of a larger memory.

The CAMBIST implemented as part of the state machine targets 64 eight-entry CAMs. Each entry has eight bits plus a valid and an active bit; both active and valid bits have their own write-enable control. The CAM comparators consist of 10 XNORs that feed an AND gate. The comparator logic can only be tested by applying patterns through the CAM memory, ie. the contents of the CAM cannot be directly observed or read. The CAMBIST circuitry in the state machine first generates a pattern to verify the AND gate by forcing all XNOR outputs to a 1; it then follows with ten patterns which disable the XNORs one at a time. Testing the comparator requires 18 input combinations: all 1s, all 0s, each one of eight on, and each one of eight off. Expect data for the CAM outputs are calculated for the same 18 combinations. During normal operation, only one of the eight outputs is active; but, during CAM testing, a 1 is often expected on all eight outputs, to shorten test time. The last pattern writes unique data in each CAM location and then verifies the CAM contents; the data is retained during subsequent testing so that the CAM can supply bit-address selection to the RAM.

### Summary

A built-in self-test(BIST) state machine has been fabricated which operates at 370 MHz, thoroughly tests various on-chip memories with deterministic patterns,

and stores failing addresses for redundancy implementation; flexibility was incorporated to ensure high-quality testing. A content addressable memory BIST was also implemented for testing both CAM logic and CAM memory embedded with the static RAM.

### Acknowledgments

The authors thank the people who characterized and digitized this circuitry, especially T. Frederick, S. Hall, G. Holsopple, J. Milinchik, S. Moran, R. Newman, M. Robillard, and D. Strandberg.

### References

- [1] R. Dekker, F. Beenker, L. Thijssen, "A realistic self-test machine for static random access memories," 1988 International Test Conference, pp. 353-361.
- [2] K. Yokomizo, K. Naito, "A 333 MHz, 72 Kb BiCMOS pipelined buffer memory with built-in self test," 1992 Symposium on VLSI Circuits Digest of Technical Papers, pp. 32-33.
- [3] A. Krasniewski, K. Gaj, "Is There Any Future for Deterministic Self Test of Embedded RAMs?" IEEE Computer Society Press, 1993, pp. 159-68.
- [4] L.G. Heller, W.R. Griffin, J.W. Davis, N.G. Thoma, "Cascode voltage switch logic: a differential CMOS logic family," 1984 IEEE ISSCC, pp. 16-17.
- [5] K.M. Chu, D.L. Pulfrey, "A comparison of CMOS circuit techniques: Differential cascode voltage switch logic versus conventional logic," IEEE J. Solid-State Circuits, Vol. 22, No.4, August 1987, pp. 528-532.
- [6] W. Daehn, J. Gross, "A test generator IC for testing large CMOS- RAMs," 1986 International Test Conference, pp. 18-24.
- [7] H. Bonges, R.D. Adams, et al., "A 576K 3.5-ns access BiCMOS ECL static RAM with array built-in self test," IEEE J. Solid-State Circuits, Vol. 27, No. 4, April 1992, pp. 649-56.
- [8] J.H. Dreibelbis, E.L. Hedberg, J.G. Petrovick, Jr., "Built-in self test for integrated circuits," US patent #5,173,906.