

Self-testing the 16-Mbps adapter chip for the IBM token-ring local area network

by S. F. Oakland
J. L. Corr
J. D. Blair
R. D. Norman
W. J. DeGuise

This paper describes the boundary-scan and built-in self-test (BIST) functions of the IBM token-ring local area network (LAN) adapter chip. These functions present a number of unique features. First, less than 1% of available standard cell circuits were needed to implement these functions. Second, clocking methods used in different logical macros were merged into a comprehensive clocking sequence for self-test. Finally, asynchronous serial and parallel interfaces were provided to facilitate the communication between a test system and the chip's built-in test circuits. Although self-test and boundary-scan provide for an inexpensive higher-level package test, evaluation showed that automatically generated deterministic patterns provide a better-quality VLSI chip manufacturing test.

Introduction

The 9.02×9.02 -mm LAN [1] adapter chip [2] was designed to be a common adapter for a variety of devices within a single physical location. It was built in $1\text{-}\mu\text{m}$ CMOS technology with two levels of metal and an additional mask level for fabricating capacitors. The chip includes 24K circuits of standard cell [3] gates, a custom microprocessor macro consisting of 10K circuits, and an analog macro. On-chip circuits operate at nine different clock rates ranging in frequency from 2.67 MHz to 32 MHz, interfacing to a LAN operating at either 4 or 16 megabits per second (Mbps). Of the approximately 1100 circuits added for boundary-scan and self-test, about 900 were built into input/output (I/O) circuits without increasing the size of the standard I/O cell. Only ~200 self-test circuits were added in the standard cell portion of the chip.

To ensure that boundary-scan and self-test circuits have negligible impact on chip performance, size, or function, digital I/O circuits were customized to include shift register latches (SRLs), called "boundary latches." All boundary latches are connected into a single scan chain for serial access. Externally applied stimuli at all primary input (PI) receivers can be captured in corresponding input boundary latches. Data stored in output boundary latches can be applied to the primary output (PO) pins via off-chip driver circuits.

©Copyright 1990 by International Business Machines Corporation. Copying in printed form for private use is permitted without payment of royalty provided that (1) each reproduction is done without alteration and (2) the *Journal* reference and IBM copyright notice are included on the first page. The title and abstract, but no other portions, of this paper may be copied or distributed royalty free without further permission by computer-based and other information-service systems. Permission to *republish* any other portion of this paper must be obtained from the Editor.

While boundary latches facilitate testing I/O circuits and circuit-card interconnections [4], they are also necessary for self-test of internal circuits. Boundary latches connected to the PIs are configured as a PRPG, and boundary latches connected to POs are configured as a MISR for self-test. SRLs in bidirectional I/O (BIO) circuits are used for stimulation and observation of internal circuits.

Boundary latches, in conjunction with an "auto-load" test mode, also provide a flexible interface to an external test system. This interface is described in a later section on chip initialization for self-test.

This paper is organized into two major sections. The first section, on test architecture, gives a brief description of several test and design verification features built into the adapter chip. The second section, on self-test, describes the details of the adapter chip's built-in self-test and auto-load modes.

Test architecture

As indicated above, this device has a mixture of semi-custom analog and digital circuits, including a built-in microprocessor that interfaces directly with a token-ring LAN. Because of the complexity of the device and the variety of products into which it may be introduced, a number of different capabilities were developed to aid system integration and testing at all levels.

In addition to the level-sensitive scan design (LSSD), a number of other, somewhat unique, test capabilities exist, including microprocessor isolation, analog macro isolation, boundary-scan, auto-load, and self-test. A set of three PI signals selects one of eight "modes," including these test/isolation modes. For reference, each of these modes is explained briefly below

• Microprocessor isolation

It was deemed necessary, very early in the project, to provide a microprocessor isolation mode for verification test and debug purposes. During normal manufacturing test, the microprocessor is tested in the standard LSSD dc stuck-fault test performed for the entire product. However, a special mode was developed for design verification that allows the microprocessor to be electrically cut off from the remainder of the chip and controlled by an entirely separate set of clock signals. These clocks include special LSSD test clocks as well as special system and cache memory clocks.

• Front-end macro (FEM) isolation

The adapter design provides for a number of LAN interface methods. Because of its front-end isolation capability, the device can be connected to the LAN either through the on-chip analog front-end macro or through an external front-end module. This isolation capability is

also used during the testing of both analog and digital circuits associated with the front-end macro. Testing of this analog macro is explained in a companion paper [5].

• Boundary-scan

One of the test techniques which is particularly useful in testing at the second and higher levels is boundary-scan [4, 6]. The general implementation of boundary-scan employs latches which are connected to each of the device I/Os, and also serially in a scan chain. This allows the user to test the product in a reduced-I/O approach by scanning test data into inputs and scanning test results out the output side [7, 8]. It also provides the second-level test with an electronic-chip-in-place test (ECIPT) capability for testing device interconnection as well as device functionality. Boundary-scan includes three submodes: LOAD, DRIVE, and SCAN.

LOAD submode is used to load all of the primary inputs and bidirectional I/O latches in parallel. This function is good for receiving second-level continuity tests, using the ECIPT concept, wherein data at the PI pin are captured by the corresponding boundary latch. It also can be used to load self-test "seeds" and initialization data, as described later.

DRIVE submode is used to apply data to the chip PO and BIO pins in parallel. This function can also be used in ECIPT to drive data to the next package, and it can be used to output the self-test results from the MISR, as described below.

SCAN submode is used to scan data serially through the boundary latches. In this mode, the boundary latches act in the same manner as a normal scan chain. However, no internal latches are clocked in this mode. Only the scan-out drivers are enabled here; all other PO drivers are disabled. This mode can be used in conjunction with either the LOAD submode to scan out the received data, or with the DRIVE submode to scan in the drive data.

• Auto-load

Auto-load mode provides a unique method of loading internal scan strings. Data at the PIs (and BIOs) are strobed into the boundary latches in parallel and then serially scanned into the internal LSSD scan strings. There are eight PIs and BIOs for each of the eleven internal scan strings. Data are loaded in parallel into these 88 boundary latches, then shifted eight bits at a time into each of the eleven internal scan chains. This sequence is then repeated (26 times) until all of the internal chains are loaded with the input data. Auto-load mode can be used with either LSSD or oscillator clocking. When the LSSD clocks are used, all clock timing, counting, etc., must be controlled by the tester. When the oscillator input is used, all timing, counting,

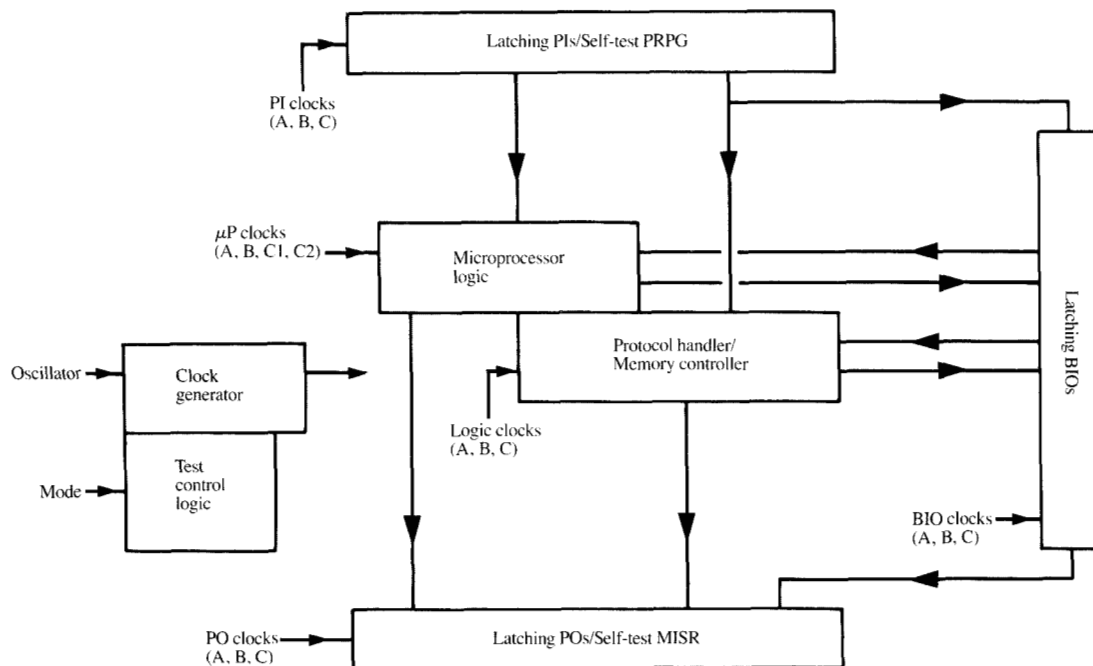


Figure 1

Logic structure of the adapter chip.

and controls are automatic. This mode can be particularly helpful in loading the initialization and "seed" data for the self-test mode, described below.

• Self-test

The self-test mode has been designed to allow the adapter chip to be tested using a minimum number of primary I/O and requiring little or no test generation. Pseudorandom test data are generated automatically by the device itself, using a linear feedback shift register (LFSR). A resulting "signature" is then stored in a multiple-input signature register (MISR).

Self-test is clocked by a single oscillator input. All clock timing, counting, and controls are automatically generated by the device. Internal LSSD registers are loaded in blocks of 212 (length of the longest scan chain) from the LFSR. Tests are performed in integral groups of 256.

Initialization of all memory circuits to a known state must be accomplished before self-test can be performed. This function can be performed using the LSSD scan

chains, the auto-load mode, or a combination of boundary-scan and auto-load modes. Initialization is described in more detail subsequently.

A detailed description of the self-test methodology is given in the next section.

Self-test

A conceptual view of the logic structure of the digital portion of the adapter chip is illustrated in **Figure 1**. This includes all parts of the chip except the front-end macro, described above, from which it is electrically isolated. All logic depicted here has been designed according to the IBM LSSD [9] design rules. The logic can be broken down into three large categories: the microprocessor, the protocol handler, and the memory controller. In addition, for the purpose of self-test, special clock generator, test control logic, and I/O circuit configurations have been designed, as shown.

At the top of Figure 1, latches in the 30 primary input circuits, configured as a LFSR, form a 30-bit PRPG for self-test. A pseudorandom stream of 1s and 0s is shifted

directly into the scan chains, thereby applying pseudorandom bits to the combinational logic. It is of interest to note that the pseudorandom data are applied to the PI-to-PO, PI-to-latch, and latch-to-PO logic, not merely to the latch-to-latch combinational logic. Special clocking for the PRPG [indicated as "PI clocks (A, B, C)"] is generated by the clock generator and test control circuitry.

At the bottom of Figure 1, latches in the 18 primary output circuits are configured as an 18-bit multiple-input signature register (MISR) for self-test. The response of the combinational logic structure is clocked directly into the MISR and also into the scan chains. While a new pseudorandom test vector is shifted into the scan chains, the response of the previous vector is compressed into the MISR. The MISR collects data from POs and BIOs as well as scan chains, thereby substantially increasing testability coverage. Special clocking for the PO/MISR is indicated in the figure as "PO clocks (A, B, C)"; this is generated by the clock generator according to the rules of the test control logic.

At the right side of Figure 1, bidirectional input/output (BIO) circuits allow an I/O pin to be used either as an observability point (output) or as a controllability point (input). During self-test this observability/controllability function is shifted from the I/O pins to latches within the BIO circuits. The BIO scan chain is filled with a pseudorandom stream of bits from the PRPG at the same time that the internal scan chains are loaded with pseudorandom data. Special clocking for the BIO circuits, shown in the figure as "BIO clocks (A, B, C)," is generated by the clock generator with special rules supplied by the test control logic.

The clock generator, shown on the left side of Figure 1, supplies all clocking for self-test (and normal system operation). All internal clocks are generated from a single oscillator input. About 200 circuits control the data flow and the sequence of internal clock pulses generated during self-test. The logic required to determine when each of these clock signals is to be used is contained in the test control logic.

The test control logic, described in detail below, is composed of counters and other logic used to control the clock generator, establishing the proper timings for self-test.

The greater portion of the logic, the protocol handler and the memory controller, was designed using a VLSI design system [3, 10]. Although this part of the logic also includes a significant amount of embedded RAM, it uses standard LSSD double-latch-design SRLs. Clocking for this part of the logic is designated in the figure as "Logic clocks (A, B, C)."

The microprocessor logic was designed using LSSD two-phase, single-latch SRLs. Clocking for this portion of

the logic is indicated as " μ P clocks (A, B, C1, C2)," as shown. The C1 and C2 clocks are the two clocks generated for normal system operation.

• *Circuit structures*

Even though the chip was designed according to LSSD rules, a variety of design styles were used by various design teams. A number of unique circuits were used internally, as well as for the boundary I/O. The following subsections describe some of these circuits, including LSSD double-latch design, LSSD single-latch design, embedded memory arrays, PIs, POs, and BIOs.

LSSD double-latch design

The majority of the chip logic was designed with a double-latch LSSD methodology, as depicted in Figure 2. Logic A clock is used only for serial scan. During normal operation, logic C clock and logic B clock are alternately activated to latch signals in the SRL. During self-test, logic C clock is activated to load the response of a pseudorandom test vector into the L1 (master) latch, and logic B clock is pulsed to copy the response data into the L2 (slave) latch. Logic A clock and logic B clock are alternately toggled to unload the data through the scan path to the MISR while loading the SRL with a fresh pseudorandom bit from the PRPG.

LSSD single-latch design

One portion of the logic, a microprocessor macro, was designed with a two-phase single-latch LSSD methodology as depicted in Figure 3. During normal operation, μ P A clock and μ P B clock are not used; μ P C1 clock and μ P C2 clock are alternately activated to latch separate signals in L1 (master) latch and L2* (slave) latch, respectively. During self-test, a pseudorandom bit is loaded into the L1 and L2* portions of the SRL through the scan path. The response of combinational logic B to the pseudorandom test vector is clocked into L1 latches with μ P C1 clock. Those response data are used as a test vector for combinational logic A, whose response is captured in L2* latches with μ P C2 clock. Microprocessor A clock and μ P B clock are alternately pulsed to shift the L2* response data into the MISR. At the same time, the next pseudorandom test vector is shifted into the latches via the scan path.

Embedded RAMs

Embedded random-access memories (RAMs) pose special problems for self-test. Either the RAMs must be initialized to known values before self-test, or the RAM outputs must be forced to known values during self-test. With the latter alternative, embedded RAMs are not tested; they are merely inhibited from affecting self-test of combinational logic. We chose to initialize RAMs and include them in the self-test.

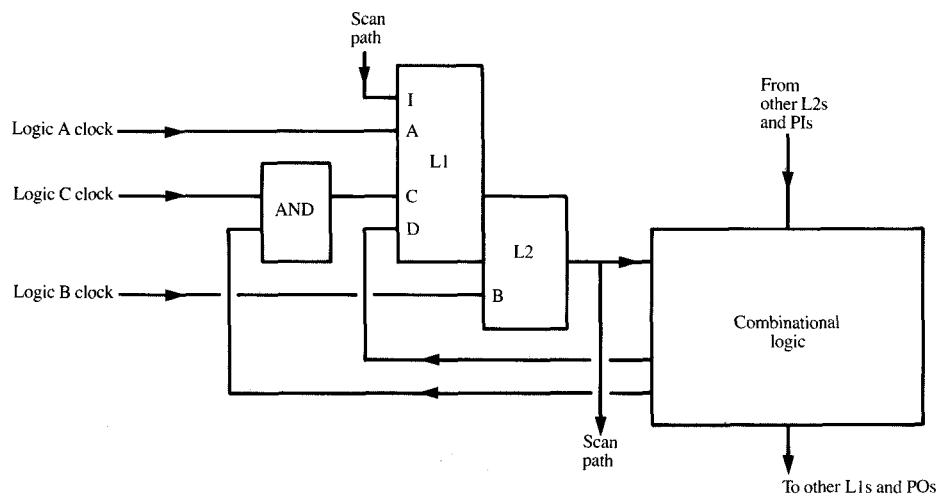


Figure 2

LSSD double-latch design.

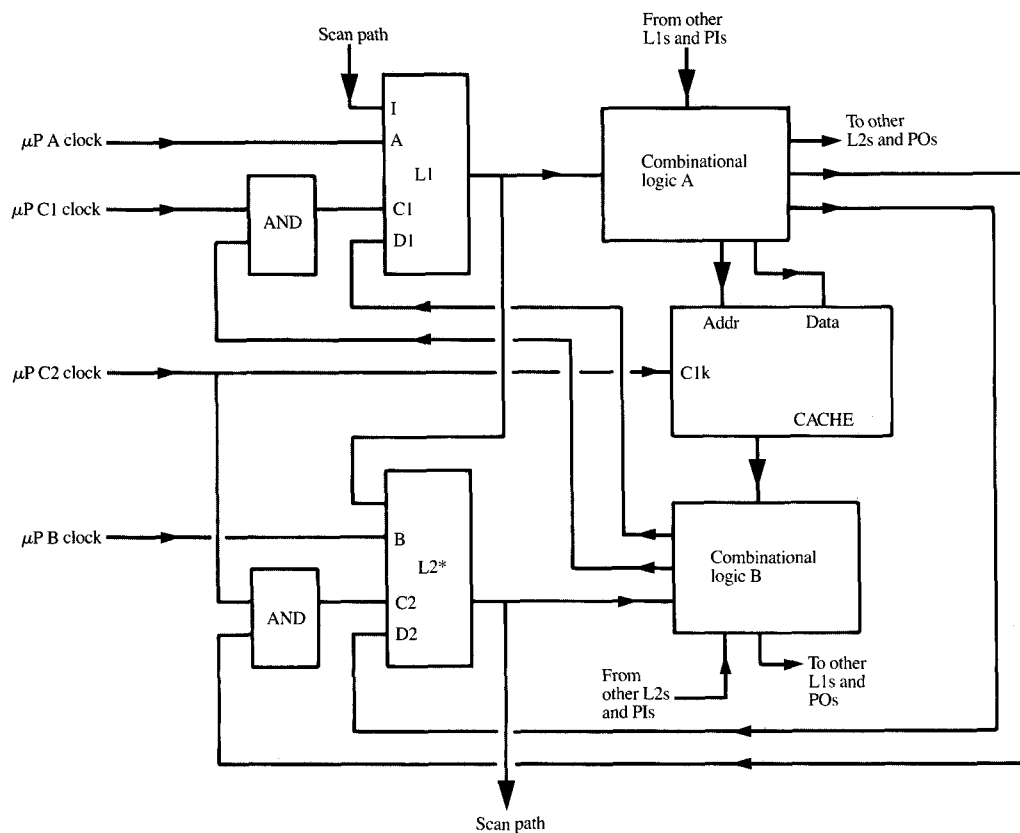


Figure 3

LSSD two-phase single-latch design with embedded cache array.

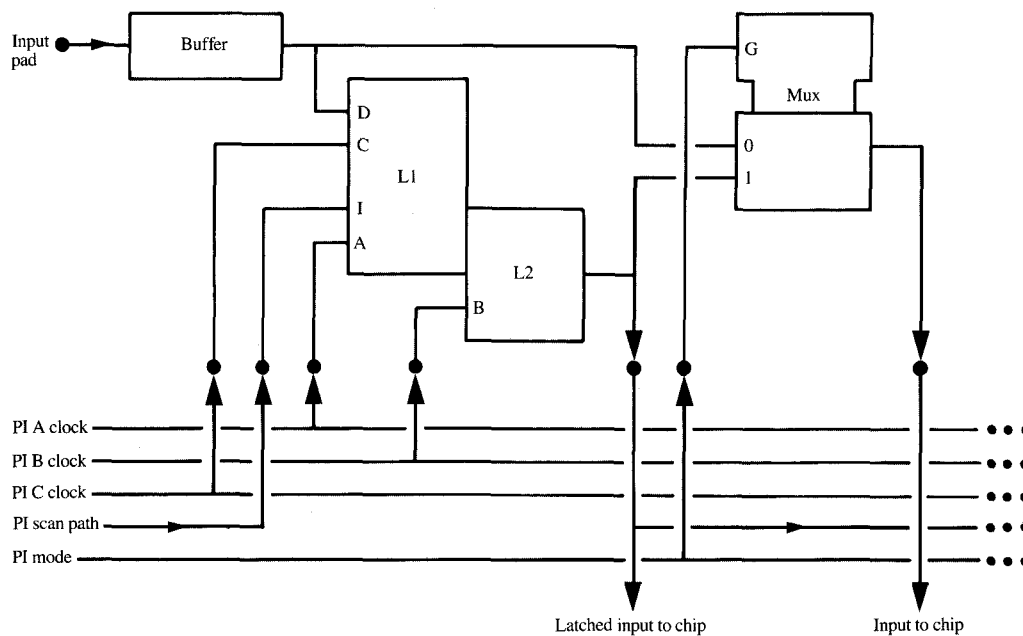


Figure 4

Latching primary input circuit.

One embedded RAM is the cache shown in Figure 3. During self-test, as in normal operation, data are written into the RAM only when μP C2 clock is active. The figure points out why embedded RAMs generally make it difficult to obtain high test coverage: (1) Combinational logic A is not fully tested except by storing some response data in the cache and then propagating those data through combinational logic B to a shift register latch. (2) Combinational logic B is not fully tested unless appropriate stimuli have already been stored in the cache. (3) The RAM itself is not fully tested unless combinational logic A allows the appropriate set of stimuli to propagate to the RAM inputs and combinational logic B allows the RAM outputs to be sufficiently observable at scannable latches.

However, it is expected that embedded RAMs on this adapter chip will have a negligible impact on the quality of self-test. There are two reasons for this:

1. Each of the embedded RAMs is relatively small; a total of 1264 bits is distributed among six RAMs. RAM area is a small percentage of total chip area.
2. There is little combinational logic between SRLs and embedded RAM I/O. Therefore, the RAM inputs are

highly controllable from SRLs, and RAM outputs are highly observable by SRLs.

Embedded RAMs must be initialized to a repeatable state before self-test. The initialization process is simplified by the fact that all address, control, and data input registers are accessible via the scan paths. RAM initialization data can be loaded into the scan paths through the parallel LSSD scan chains or via the auto-load feature. Chip initialization is discussed in more detail subsequently.

PI circuits

Requirements for boundary-scan and self-test led to the inclusion of a shift register latch in each PI circuit. This latch was added primarily for testing, but it was also made available to the logic designers for functional use.

A latching PI circuit is shown in Figure 4. When PI mode signal is zero, a data path is enabled from the chip input pad to the interior of the chip. The signal on the input pad may be latched in the shift register latch for later use by the chip's internal logic. For self-test, PI mode is held at 1, disabling the data path from the input pad to the chip's internal logic. Furthermore, to prevent external signals from affecting SRL contents, PI C clock is never activated during self-test.

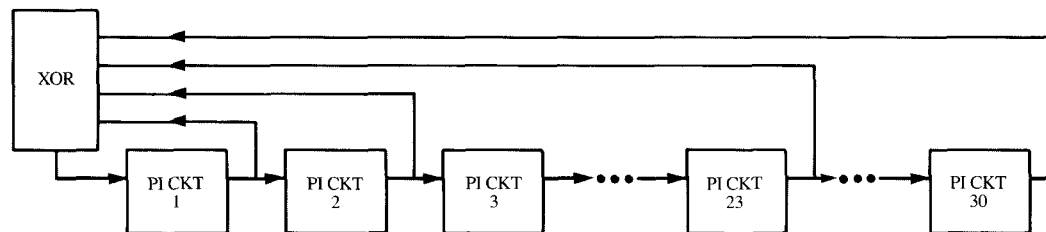


Figure 5

Pseudorandom pattern generator.

For self-test, thirty PI circuits are connected together into a 30-bit PRPG, as shown in **Figure 5**. When PI A clock and PI B clock are toggled, a pseudorandom bit is shifted into PI circuit 1 while the bit in PI circuit n is shifted through the scan path into PI circuit $n + 1$. The PRPG LFSR implements the primitive polynomial $1 + x^7 + x^{28} + x^{29} + x^{30}$.

PO circuits

Requirements for self-test and boundary-scan led to the inclusion of a shift register latch in each PO circuit. A latching primary output circuit is shown in **Figure 6**. For normal operation the signal driver data select is zero, allowing output data from the chip to propagate to the output pad whenever the three-state driver (TSD) is enabled. The SRL is used only for boundary-scan and self-test. During self-test, output data from the chip are "exclusive-ORed" with PO scan path, the data in the SRL of another primary output circuit. The output of the exclusive-OR gate is latched by either PO A clock or PO C clock. PO B clock shifts the data from the L1 latch into the L2 latch.

For self-test, eighteen PO circuits are connected together to form an 18-bit MISR, as shown in **Figure 7**. When the signal scan mode is zero, output data from the chip are compressed into each MISR SRL as described above. On the other hand, if scan mode = 1, only data from scan chains are shifted and compressed into the signature register. The MISR LFSR polynomial is $1 + x^9 + x^{18}$.

BIO circuit

An SRL was added to each bidirectional I/O circuit to satisfy the self-test and boundary-scan requirements. A BIO circuit is shown in **Figure 8**. When the three-state driver is disabled, external signals may be applied to the

I/O pad. If inbound select = 0, external input signals propagate to the chip interior; otherwise the chip input is sourced from the circuit's SRL. During self-test, inbound select is held at 1, disabling the external input.

Output data to the I/O pad are controlled by the outbound select signal. When outbound select = 0, data from the chip interior may be driven onto the I/O pad and/or latched into the SRL. When outbound select = 1, SRL data may be driven onto the I/O pad. During self-test, outbound select is held at 0; BIO C clock is pulsed to capture the response of the chip logic in the boundary latch. BIO A clock and BIO B clock are switched to unload the BIO scan chain into the MISR while loading it with pseudorandom data from the PRPG.

• *Self-test operation*

Before self-test, the adapter chip must be initialized. Once initialized, self-test is controlled by a small state machine.

Initialization

For the self-test signature to be repeatable, the adapter chip must be initialized to a repeatable state. Every shift register latch and every bit of every embedded RAM array must be initialized.

Embedded RAMs are initialized by scanning address, data, and control bits into appropriate registers and then applying external clocks to the chip, writing data into the specified address of each RAM. This process is repeated until all RAM addresses have been written.

Several options exist for scanning initialization data into the chip. The chip's 12 scan chains can be loaded from the 12 scan inputs, or the chip's auto-load mode can be used. If the auto-load mode is used, data are first loaded into the chip's 11 input registers. Two options exist for loading these eight-bit registers: (1) boundary-scan mode may be used to serially load the input latches,

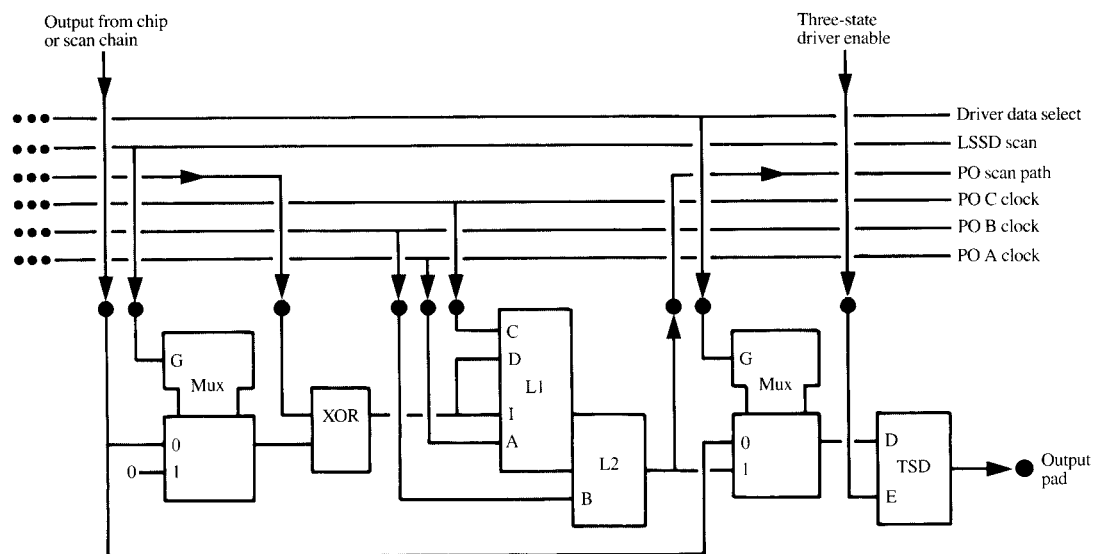


Figure 6

Latching primary output circuit.

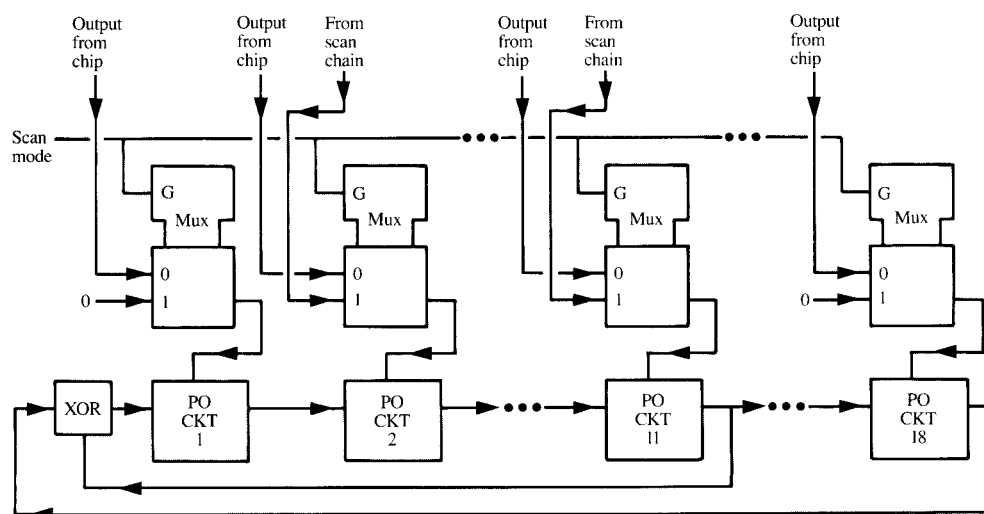


Figure 7

Multiple-input signature register and scan path multiplexing logic.

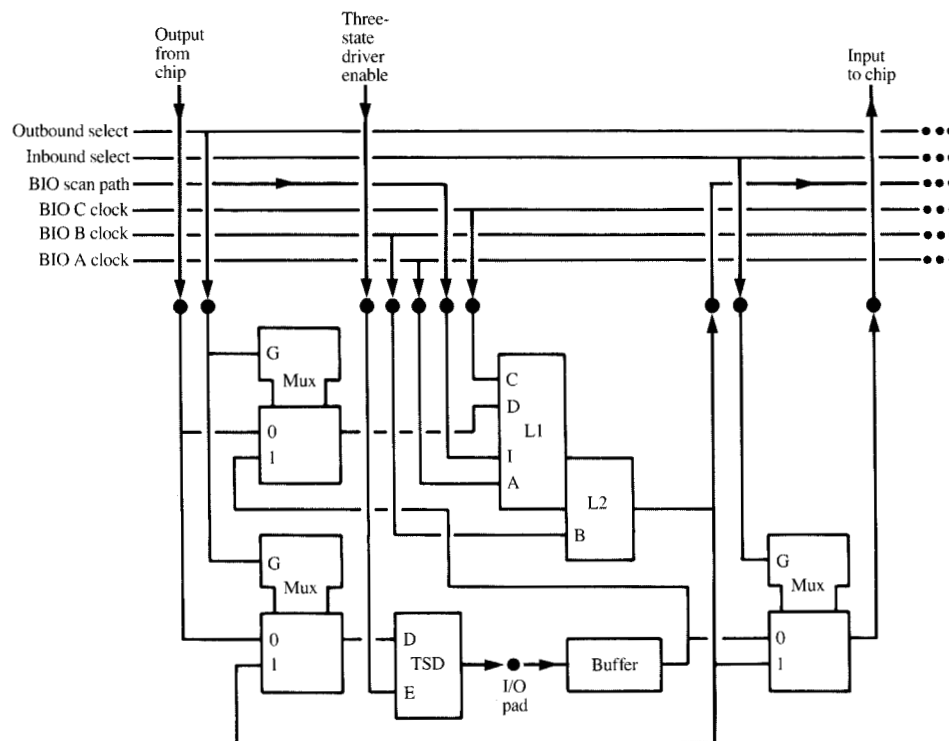


Figure 8

Latching bidirectional input/output circuit.

or (2) all input latches may be loaded simultaneously from their corresponding input pins, using either boundary-scan mode or auto-load mode. Once the input registers are loaded, auto-load mode "automatically" shifts the eight bits of data from each input register into the corresponding internal scan chain.

Operation of the auto-load mode is discussed in more detail in the following section.

Self-test control

Self-test is controlled by a four-bit state machine, two counters, and some data-path and clock-gating logic. When GO, a nonlatched primary input signal, is activated, the state machine initiates a self-test sequence. Because the first test pattern is not a pseudorandom pattern, but rather the initial state of the chip, all shift register latches and all embedded arrays must be initialized to known values, as indicated above, before GO is activated.

The state machine is implemented as 12 states of a four-bit register, which controls both self-test and auto-load. As the register conditions progress through the various states, signals are sent to the clock generator, the test counter, and the scan counter to control the test operation. The state diagram is shown in **Figure 9**. Each state is described in the following subsections.

State 0 (Idle) This is the normal state of the machine when neither self-test nor auto-load is active. Entry into the Idle state automatically reloads the test counter. Self-test is begun by engaging the self-test mode and then activating the GO PI. This causes the state machine to switch to State 1 (Measure). Auto-load is begun by engaging the auto-load mode, causing the state machine to switch to State 7 (Boundary B).

State 1 (Measure) This state causes the test counter to be decremented and a "measure" to occur. In self-test

State 9 (Scan) During self-test this state causes all data stored in scan strings to be shifted into the MISR. Simultaneously, new data are generated by the PRPG and shifted into the scan strings, until the scan counter reaches zero. The A/B LSSD scan clocks required to serially shift the PRPG, MISR, and all of the scan chains are generated iteratively until the scan counter reaches zero.

Similarly, in auto-load mode, data from input registers are shifted into the internal scan strings until the scan counter reaches zero. The system moves next to State 10 (Post-scan).

State 10 (Post-scan) During self-test this state is used to apply the final LSSD B clock to the PRPG, MISR, and the scan chains. If the test counter is zero at this point, the system goes back to State 0 (Idle). However, if the test count is not zero, the system goes immediately to State 1 (Measure). Since the test counter is initially set to 256, the system makes 256 cycles through States 1–10 before returning to State 0 (Idle). In auto-load mode the final B clock to the input registers and scan chains occurs during State 10. Next, the system moves to State 11 (AL-wait).

State 11 (AL-wait) In the auto-load mode this is a "wait" or "idle" state, in which no operation occurs. The state machine stays in this state as long as auto-load mode is active and the GO signal is not activated. If the GO signal is activated while in auto-load mode, the state machine moves to State 1 (Measure). If auto-load mode is deactivated, the system moves to State 0 (Idle).

If, during self-test, more than 256 test loops are required, the state machine can be restarted from State 0 (Idle) merely by activating the GO signal. Since the accumulated MISR data are not destroyed following each 256-test sequence, restarting the test merely continues where it left off. The length of the test is therefore dependent only upon the desired coverage. The longer the test is run, the better the coverage. The MISR can be checked at any or all of the available idle states during the test.

In addition, a new seed or an entire test can be scanned in when the system reaches State 0 (Idle). This allows the user to test specific random-pattern-resistant faults by inserting nonrandom patterns into the test.

• Evaluation

To study self-test effectiveness, a random subset (5%) of stuck faults was selected for fault simulation. Rather than simulating scan operations, SRLs were directly "stimulated" and "measured." SRL stimulation data were computed by calculating PRPG output data for each scan cycle.

A fault-simulation tester loop consisted of stimulating the SRLs, then stimulating the clocks in the sequence described by States 1–7 above. If a fault was propagated to an L2 latch, it was assumed detected, because it would be shifted into the MISR during the self-test scan state. Possible masking effects in the MISR were not considered.

Based on the results of simulation of the random subset of faults, predictions were made about the coverage of the entire set of stuck faults. With 256 pseudorandom patterns, fault coverage is 80%. With ten sets of 256 pseudorandom patterns, fault coverage increases to 85%.

At the time the fault simulation was performed, the authors had limited tools for analyzing untested faults. Perhaps some were random-pattern-resistant; perhaps some would be tested if States 1–7 of the self-test controller were sequenced in a different order; perhaps some would have been tested with additional pseudorandom patterns. However, the cost of further simulation was prohibitive. Furthermore, no effective method was found for self-testing the analog macro.

In contrast, automatically generated LSSD patterns tested 97.5% of logic stuck faults, and tests were manually generated for the analog macro. For these reasons, deterministic stored patterns, rather than self-test, are being used in the manufacture of the LAN adapter chip.

The LSSD boundary-scan architecture was invaluable in debugging the chip. The architecture allows the chip operation to be halted, internal latches to be unloaded, incorrect bits to be corrected and reloaded [11], and normal chip operation to be resumed.

Summary

Test features of the IBM token-ring LAN adapter chip have been presented. Boundary-scan and self-test circuits have been described. Chip initialization alternatives were discussed. Finally, self-test stuck-fault coverage was evaluated, and operation was verified in hardware. Because self-test fault coverage was not demonstrated to be better than that of deterministic stored patterns, self-test is not used for chip manufacturing test.

References and note

1. IEEE Project 802, "ISO 8802/5 Token-Ring Local Area Network Standard," Institute of Electrical and Electronics Engineers, Inc., New York, 1985.
2. K. W. Lang, C. K. Erdelyi, J. L. Lamphere, S. F. Oakland, J. D. Blair, A. Correale, Jr., H. C. Cranford, D. A. Dombrowski, C. R. Hoffman, J. K. Lee, J. M. Mullen, and V. R. Norman, "A 16-Mbps Adapter Chip for the IBM Token-Ring Local Area Network," *Proceedings of the 1989 IEEE Custom Integrated Circuits Conference*, pp. 11.3.1–11.3.5.
3. A. W. Aldridge, R. F. Keil, J. H. Panner, G. D. Pittman, and D. R. Thomas, "A 40K Equivalent Gate CMOS Standard Cell Chip," *Proceedings of the 1987 IEEE Custom Integrated Circuits Conference*, pp. 248–251.

4. P. T. Wagner, "Interconnect Testing with Boundary Scan," *Proceedings of the 1987 IEEE International Test Conference*, pp. 52-57.
5. J. J. Van Horn, R. A. Waller, R. J. Prilik, and K. C. Bocash, "LAN Interface Chip and Mixed-Signal Testing Developments," *IBM J. Res. Develop.* **34**, 428-441 (1990, this issue).
6. Proposed IEEE P1149.1, Standard Test Access Port and Boundary-Scan Architecture, Draft D5, Institute of Electrical and Electronics Engineers, Inc., New York, June 20, 1989.
7. R. W. Bassett, S. L. Dingle, P. S. Gillis, J. G. Petrovick, B. J. Butkus, M. R. Faucher, J. H. Panner, and D. L. Wheeler, "Low Cost Testing of High Density Logic Components," *Proceedings of the 1989 IEEE International Test Conference*, pp. 550-557.
8. R. W. Bassett, M. E. Turner, J. H. Panner, P. S. Gillis, S. F. Oakland, and D. W. Stout, "Boundary-Scan Design Principles for Efficient LSSD ASIC Testing," *IBM J. Res. Develop.* **34**, 339-354 (1990, this issue).
9. E. B. Eichelberger and T. W. Williams, "A Logic Design Structure for LSI Testability" *J. Design Aut. & Fault-Tol. Computing* **2**, 165-178 (May 1978).
10. R. Hornung, M. Bonneau, B. Waymel, J. Fiore, E. Gould, R. Piro, J. Martin, L. McAllister, and S. Tom, "A Versatile VLSI Design System for Combining Gate Array and Standard Cell Circuits on the Same Chip," *Proceedings of the 1987 IEEE Custom Integrated Circuits Conference*, pp. 245-247.
11. In the microprocessor some L2* latches function independently of their corresponding L1s, but cannot be loaded independently via the scan path.

Received May 18, 1989; accepted for publication November 6, 1989

Steven F. Oakland *IBM General Technology Division, Essex Junction, Vermont 05452.* Dr. Oakland received his B.S. and M.S. degrees from South Dakota State University in 1973 and 1974, respectively. He received his Ph.D. degree in electrical engineering and biomedical engineering from Iowa State University in 1980, joining IBM that same year at Rochester, Minnesota. While at Rochester, he was involved with computer display terminal development. In 1983, Dr. Oakland joined the faculty of the Department of Computer Science, State University of New York, Plattsburgh, as a temporary assistant professor. Since 1984, he has been associated with the IBM Burlington development laboratory in Essex Junction, where he is currently an advisory engineer in the microsystem products organization. His principal activities and interests are in the areas of design for testability, design verification, and test generation.

James L. Corr *IBM General Technology Division, Essex Junction, Vermont 05452.* Mr. Corr is currently an advisory engineer in the Microcomponent Tool Development Department, where he is involved in hardware design verification, test, and diagnostics strategy development. Since joining IBM in 1979, he has been involved in both the design and manufacturing of various logic devices, as well as the design and development of ASIC design systems and design system tools. Much of this work has been in the area of automated test and diagnostics systems. His work on the 16-Mbps adapter chip was specifically involved with the specification and development of the self-test mechanism. Mr. Corr holds a Bachelor of Electrical Engineering degree from the University of Minnesota and a Master of Science degree in electrical engineering from California State University.

Jack D. Blair *IBM Communication Systems Division, P.O. Box 12195, Research Triangle Park, North Carolina 27709.* Mr. Blair received his B.S. in electrical engineering from Clemson University. He is a staff engineer in Technology Development at the RTP facility. Since joining IBM in 1984, he has been involved in the development of various token-ring products. Mr. Blair's work on the 16-Mbps token-ring adapter chip includes logic design, device-level circuit analysis, gate-level simulation and timing analysis, test pattern generation, and design tool development.

Vernon (Rob) Norman *IBM Communication Systems Division, P.O. Box 12195, Research Triangle Park, North Carolina 27709.* Mr. Norman is a development engineer in the Technology Development organization in Raleigh. He joined IBM in 1979 at Raleigh, where he worked in the Communications Adapter Development group on microprocessor-based communications adapters. In 1983, he became part of the VLSI design team developing chips for the IBM token-ring network adapter products. From 1984 to 1985, he worked as lead logic designer and in 1985 became manager of the design team developing the 16-Mbps token-ring CMOS adapter chip, which was announced in 1988. Mr. Norman received an IBM Outstanding Technical Achievement Award in 1986 for his work leading to the announcement of the IBM token-ring network.

Wayne J. DeGuise *IBM General Technology Division, Essex Junction, Vermont 05452.* Mr. DeGuise is currently a senior associate engineer/scientist in the Product Engineering/Advanced Products Department. He joined IBM in 1977 at Essex Junction, working in manufacturing final test of high-performance bipolar analog devices. Later, he worked on product/process characterization of 64K RAMs and 32-bit RISC microprocessors. In 1984, Mr. DeGuise became part of the VLSI design team developing the LAN adapter chip, assuming the responsibility for the general test plan for the chip, including self-test. For the last three years he has been a technical assistant on staff to microsystems functional management.