BIST of PCB Interconnects Using Boundary-Scan Architecture

Abu S. M. Hassan, *Member, IEEE,* Vinod K. Agarwal, *Fellow, IEEE,* Benoit Nadeau-Dostie, *Member, IEEE,* and Janusz Rajski, *Member, IEEE*

***Abstract*—This paper addresses the issues of printed circuit board (PCB) interconnect testing in the context of boundary-scan architecture. Boundary-scan architecture is treated here as the framework for a PCB level built-in self-test (BIST). A novel BIST method is developed which utilizes various features of the architecture. Boundary-scan architecture is shown to have the capability to generate time efficient test vector sets. Response compaction within the boundary-scan chain is introduced to reduce shift out time as well as to simplify detection and diagnosis. However, the proposed BIST schemes require some extensions of the standard boundary-scan cells, and schemes can work only if every boundary-scan cell of every IC on the PCB has the proposed extensions.**

I. Introduction

IN RECENT years, structured design-for-testability at the printed circuit board (PCB) level has become an activity of major interest. This is a natural evolution, following a wide acceptance of the structured DFT (i.e., scan and built-in self-test) [21] at the IC level and the realization that the cost associated with implementing scan cannot be justified unless it can be used to simplify the testing efforts at the PCB and higher levels as well. This, combined with the emergence of very high density packaging technology at the PCB level, in particular, that of surface mount interconnects, and the growing interest in multichip modules, made it essential to develop the concept of boundary-scan, as detailed in [15].

The boundary-scan concept allows one to access and control the primary input and output pins on each component of the PCB from its edges. This is done by connecting all the primary inputs and outputs of each component into a shift register which has a boundary-scan input and a boundary-scan output. A simple boundary-scan cell is shown in Fig. 1 [15, fig. 1-1]. The shift registers on all the components of a PCB can be connected together to form a larger shift register with a single scan in edge and a single scan out edge, as shown in Fig. 2 [15, fig. 1-2]. A test clock line and a test mode select line, along with the boundary scan in and scan out lines, form a four-wire testability bus [15]. Thus, in effect, the boundary scan concept provides a type of electronic in-circuit testing facility.

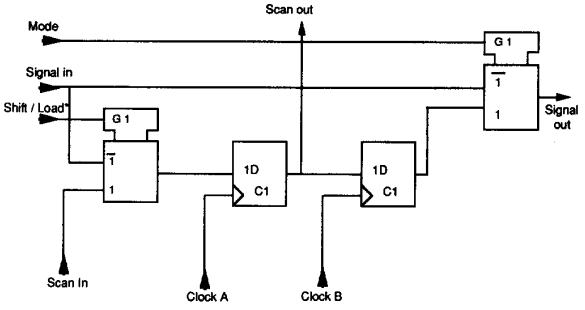


Fig. 1. A Simple boundary-scan cell.

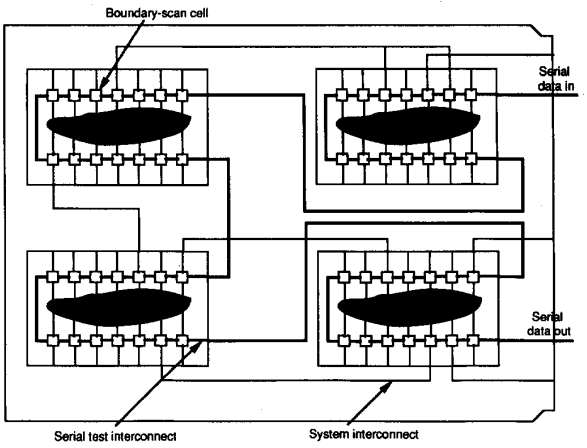


Fig. 2. Boundary-scan chain.

Using this concept at the PCB level, it should be possible to confirm that each component, such as an IC, performs its required function, and that the IC's are interconnected in the correct manner. The problem of interest in this work is that of using this concept to verify that the interconnects connecting these IC's on a PCB are free from structural faults.

The interconnection of IC's and other discrete components on a PCB is a complex maze of multilayer electrical

Manuscript received November 27, 1990; revised November 6, 1991. This paper was recommended by Associate Editor S. Seth.

A. S. M. Hassan and B. Nadeau-Dostie are with Bell Northern Research Department 5E31, Mail Stop 049, Corkstown, Ottawa, Ont. K1Y 4H7, Canada.

V. K. Agarwal and J. Rajski are with the Microelectronics and Computer Systems Laboratory, Department of Electrical Engineering, McGill University, Montréal, P.Q. H3A 2A7, Canada.

IEEE Log Number 9200812.

conductors which are likely to be failed by the presence of shorts, stuck-ats, and open faults [1], [3]. In order to test such a structure in a cost-effective way, structured techniques are required which can be easily automated and possibly implemented in a self-test environment. At the outset, it does not appear to be a simple problem until one realizes that on a single PCB there may be thousands of I/O pins all from different IC's, which are connected to each other in many different ways (unidirectional, bidirectional, one-to-many, many-to-one, forming chains and clusters, etc.). In addition, since repair at the PCB level is a necessary activity, it is not sufficient to know if the interconnect is faulty; one also must determine where the fault might be if, indeed, the PCB is faulty.

Various algorithms [6], [11], [13], [19], [22] exist in the literature for interconnect testing. Some of these algorithms were developed prior to the existence of the boundary-scan architecture. The other schemes use the boundary-scan architecture only to get easy access to the interconnect test points. The work in this paper has taken a completely different approach. A formal set of structured test generation and test diagnosis techniques are developed for interconnect faults on PCB's utilizing different properties of the boundary-scan architecture. More importantly, these techniques are efficiently implemented in a built-in self-test (BIST) manner. BIST techniques for interconnect testing are the most important contributions of this paper.

The advantages of the BIST approach at the component level are well known [21]. BIST schemes, proposed in this paper, are developed to have similar advantages at the PCB level interconnect testing. The main advantages of the proposed BIST schemes are the simple and time-efficient generation of a test vector set and a simple detection and diagnosis procedure. However, like any other BIST scheme, some overhead is involved for efficient implementation of the proposed schemes. This overhead is incorporated in the form of some extensions of the boundary-scan cells described in [15]. These extensions and their advantages are described in this paper, along with the BIST schemes.

Note that several industry implementations and papers [4], [5], [14], [17], [18] have extended the boundary-scan cells described in [15] in various ways to suit their needs. Thus, the purpose of extension of boundary-scan cells is nothing unique to the schemes developed in this work.

The BIST schemes developed in this work require that all the boundary scan cells of every IC on the PCB should have the proposed extensions. This situation is more acceptable to a vertically integrated company which designs and manufactures everything in-house, from IC's to PCB and all the way up the system. Such a company can design its boundary-scan cells with the proposed extensions and can implement BIST in a hierarchical manner to justify the cost of BIST and DFT. On the other hand, unless different ASIC vendors adopt the proposed extensions of the boundary-scan cells in their designs, the proposed schemes will have limited usefulness for a PCB where IC's are from multiple external vendors. Thus, the feasibility of the proposed schemes in the context of their design environment should be carefully reviewed.

The remainder of the paper is organized as follows. In Section II, a brief description of test access port [15] is followed by various basic concepts related to interconnects and interconnect faults. The test environment for the proposed BIST schemes is also introduced, and a very brief overview of the existing test schemes concludes this section. The proposed BIST approach, including detection and diagnosis schemes for the testing of shorts and stuck-ats in the interconnects, is described in Section III. Open fault testing issues are also addressed later in this section. Concluding remarks are made in Section IV.

II. Preliminaries

The test access port (TAP) concept of the boundary-scan architecture facilitates standard test communication protocol between IC's on the same PCB manufactured by different vendors. In [15], the TAP consists of a test data in (TDI) pin, a test data out (TDO) pin, a test clock (TCK) pin, and a test mode select (TMS) pin. These pins are used to access (i) an instruction register in TAP; (ii) the boundary-scan register; or (iii) some user defined data registers. More detail can be found in [15].

As seen in Fig. 1, the basic cell of the boundary-scan register for an input pin allows data to be either loaded into the register from the input port, or driven from the register through the output port of the cell into the core of the IC design. Similar designs of boundary-scan cells associated with output pin, bidirectional pin, and enable signal are described in [15].

Boundary-scan register cells are placed such that the state of each digital system pin (including clock pins) can be controlled or observed using the boundary-scan register. For two-state input or output pins, where signals can only be at the high (1) or low (0) logic level at any given instant, one boundary-scan register cell is sufficient to allow the state of the pin to be controlled or observed. However, for three-state pins, the capability exists for data to be driven actively or inactively such that four states are possible. Data from a minimum of two boundary-scan register cells is therefore required to allow the state (signal value plus active/inactive) of a three-state pin to be controlled or observed. For similar reasons, additional boundary-scan cells, associated with each three-state bidirectional system pin, will control whether it operates as input or output [15].

A. Structure of Interconnects

The interconnects on a PCB can have various possible structures. In order to include these different structures as units under test, and to conveniently describe the various test schemes, the term *net* is defined in the following.

Definition: A net is any group of two or more boundary-scan register I/O cells and the electrical conductors (i.e., wires) which connect all these cells into one grid.

The concept of "grid" implies that if any one conductor in a net was set to a logic value, say 0(1), then all the other conductors in that net will carry the same logic value 0(1). Each net is named after the scan cells connected to it, and each conductor in a net is named after the two endpoints of the conductor. An arrow in a net conductor indicates the direction of signal flow in that conductor. Net structures varying in number of output and input pin scan cells are shown in Fig. 3.

The testing of wired nets [8] is not considered in this paper. It is assumed that the driver associated with each output pin in a multiple output pin scan cell net has tri-state capability. During testing, tri-state is enabled/disabled by the control bit loaded in the associated boundary-scan control cell, as mentioned previously. Tri-state drivers are enabled/disabled in such a way that, at any given time, only one "selected" driver of every multiple output pin scan cell net is enabled. Such a control setting avoids the bus contention in multiple output pin scan cell nets.

The EXTEST instruction of the instruction register in TAP [15] allows the execution of an interconnect test. In a typical interconnect test scenario, boundary-scan register cells at output pins are used to apply test stimuli, while those at input pins capture test results. At output pins data shifted into the component are applied to the interconnect; at input pins data are first captured into the shift register path and then shifted out of the component for detection and diagnosis. The process of application of a test vector is explained in the following. Fig. 3(d) shows a net *QSTU* with two output pin scan cells, Q and S, and two input pin scan cells, T and U. Q and S are enabled/disabled from control cells P and Q, respectively. Suppose a logic 1 is to be applied to this net. This requires that a 1 be shifted in and loaded in one of the two output pin scan cells of this net. The loaded scan cell must be enabled from the corresponding control cell. The other output pin scan cell must be kept disabled from the corresponding control cell. In this example, let us suppose logic 1 is to be applied from Q. Therefore, as explained above, this cell is to be enabled from P (which is to be loaded with 1 as the enable control bit). S is to be disabled from R (which is to be loaded with 0) and, hence, can be loaded with don't care (0/1). The input pin scan cells T and U are to be initialized by shifting in 0's. Due to the application of 1 from Q to the net QSTU, input pin scan cells T and U should capture 1's. These output bits are to be shifted out for response analysis. Similarly, a logic 0 can be applied to a net.

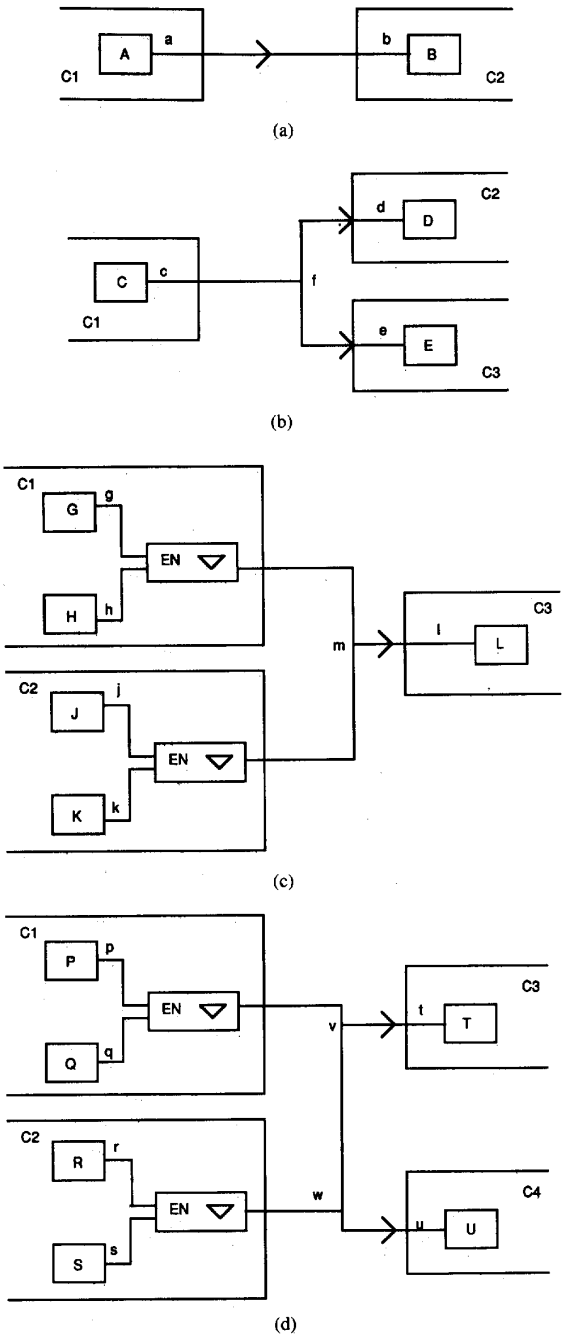


Fig. 3. (a) Simple net. (b) Net with multiple input pin scan cells. (c) Net with multiple output pin scan cells. (d) Net with cluster structure.

B. Fault Model

The fault model of interest for net testing should be based on the failures likely to be observed in the nets of PCB. As mentioned in the literature [1], [3], interconnect faults are introduced by the manufacturing processes of PCB assembly and soldering. Assembly problems result in bent, broken, or missing component leads. The soldering problems cause unwanted opens and shorts. For instance, opens can result from lack of solder, improper movement of parts, or bare printed circuit etches of feedthrough holes. Shorts are caused mainly by solder mask and solder flow.

Based on the data available in the literature [1], [3], [6], [19], interconnect faults are grouped into the following different types.

1) Short or Bridging Fault

This fault creates an (unwanted) physical connection between two or more nets. The behavior of the nets depend upon the driver characteristics of the individual nets which are shorted together. The behavior can be deterministic or nondeterministic. Deterministic behavior can be characterized as follows:

a) AND short: If the drivers of the shorted nets are such that a logic 0 dominates, then the resultant logic value is an AND of the logic values of the individual nets.

b) OR short: If the drivers of the shorted nets are such that a logic 1 dominates, then the resultant logic value is an OR of the logic values of the individual nets.

Of these two types of shorts, depending upon the technology used in the individual component, either the AND type or the OR type will occur, but not both. However, components with different technologies can be used in the same PCB. Therefore, to make the testing schemes technology independent, we will, in this paper, consider the simultaneous presence of both the AND and the OR types of shorts on a single PCB. These two short types are treated extensively in this paper.

A short between nets can create an undefined or nondeterministic logic state at the receivers of the faulty nets. The shorted outputs can be weak 0's or 1's. Analog testing is necessary to detect this type of short. Detecting the undefined faulty value is beyond the scope of the digital logic testing schemes developed in this work; therefore, this type of short is not considered in the fault model.

2) Stuck-at (One/Zero) Fault

The faulty net gets stuck to a logic value of 1 (stuck-at-one)/0 (stuck-at-zero) due to this failure.

3) Stuck-open or Open Fault

This fault creates a break or open in the faulty net. Due to the break in the net, the logic value applied from the input end does not reach the output end. However, it is possible that the output end of a net is "charged" to a logic 1(0), due to the pad protection circuitry, parasitic capacitance, etc. [7], and a logic 1(0) is applied from the input end for testing. In this case, the open fault will not be detected. Therefore, to test for an open fault, it is necessary to apply a 1–0 (0–1) transition from the input end and to observe the same transition at the output end of the net.

There is a difference between the way a short or stuck-at and an open fault affects a net. A short or stuck-at affects the entire net, whereas an open fault may affect only part of the net. Due to this nature of short or stuck-at, it is sufficient to apply test vectors from only one output pin scan cell and to monitor the outputs from only one input pin scan cell of each net during testing. In other words, the entire net can be sensitized from any output pin scan cell, and if any conductor is shorted or stuck-at, the fault effect can be observed from any input pin scan cell of the faulty net. A short is detected and diagnosed considering the net as the unit under test. No attempt is made to locate the faulty part or faulty conductor in the net.

In contrast, an open fault in any conductor of a net may not be sensitized from a single output pin scan cell and may not be observed from a single input pin scan cell of that net. For example, there are two output pin scan cells, Q and S, in net $QSTU$ as shown in Fig. 3(d). An open fault in conductor vt cannot be observed from the input pin scan cell U. In order to sensitize an open fault in every conductor in a net, input vectors should be applied from every output pin scan cell, one at a time. Also, responses should be observed at every input pin scan cell of each net. Schemes for diagnosing faulty conductors within a net are developed in [10]. However, in this paper an open fault is detected and diagnosed considering only the net as the unit under test.

The fault model assumes that both single and multiple nets can be faulty simultaneously. However, occurrence of multiple faults in the same net is not addressed here. Fig. 4 shows four nets under test. At any instance, multiple nets can be faulty, for example, AB SA-1, CD and EF OR-shorted, GH SA-0 can form the fault list. But every net, like GH, is assumed to have at most one fault. For example, faults like GH open and GH shorted to EF at the same time, are not considered in this fault model. Although some faults of this nature can be detected and diagnosed by the proposed schemes, no attempt is made in this paper for characterizing such multiple faults.

C. The Boundary Scan Environment

The following notation and definitions describe the test environment used in this paper.

N	Length of the boundary scan chain. This is the total number of input pin scan cells output pin scan cells, and control cells in the boundary-scan chain.
$N(i)$	ith scan cell in the boundary scan chain (where $i = 1$ corresponds to the boundary-scan register cell closest to the scan-in edge of the PCB).
n	Number of nets under test, $n < N$.
$n(i)$	ith net under test (where $i = 1$ corresponds to the net connected to the input scan cell closest to the boundary scan-in edge of the PCB), assuming a ring-like connection of all boundary-scan cells and one boundary-scan chain in the entire PCB.
$p(i)$	Degree of ith net. This is the total number of output pin scan cells connected to the ith net.
P	max $(p(i))$.

The following definitions are taken from [11].

- **Parallel Test Vector (PTV)** is the test vector applied to all nets of an interconnect network in parallel. Table I shows a set of input vectors $V(1)$, $V(2)$, $V(3)$, and $V(4)$. Note that each vector is to be padded with X's and O's (1's) so that "actual" data bits (shown in Table I) are loaded in the appropriate output pin scan cells and control cells are loaded with required control bits.
- **Sequential Test Vector (STV)** is the vector applied to a net over a period of time by a number of PTV's.

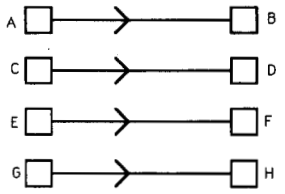


Fig. 4. Nets under test.

TABLE I
TEST SET OF log (n) VECTORS

Nets	V(1)	V(2)	V(3)	V(4)
n(1)	0	0	0	0
n(2)	0	0	0	1
n(3)	0	0	1	0
n(4)	0	0	1	1
n(5)	0	1	0	0
n(6)	0	1	0	1
n(7)	0	1	1	0
n(8)	0	1	1	1
n(9)	1	0	0	0

Different rows of Table I show the STV's applied to different nets—$n(1)$, $n(2)$, $n(3)$, etc.

- **Sequential Response Vector (SRV)** is the response of a net to an STV. If the net is fault-free, its STV and SRV will be identical. A faulty net will have differing STV and SRV.

D. Existing Test Schemes

Existing interconnect test schemes are here described very briefly. It has been shown in [13] that a set of $\log(n)$ (in this paper, all the logarithms are of base two) vectors is necessary and sufficient to detect every possible short in a network of n unconnected terminals. This set of $\log(n)$ vectors is used in [6] and [19] for detecting shorts in n nets. The diagnosis scheme described in [19] requires $2 \log (n + 2)$ vectors for a system of n nets under test. The min-weight diagnosis algorithm proposed in [22] requires p vectors, where vectors are generated by sequentially assigning a unique p-bit STV of minimum weight (number of 1's) to each of the n nets. In the max-independence diagnosis algorithm [22], test vectors are generated using net adjacency or wire routing information, as well as information about the maximum size of expected shorts or solder defects. A number of adaptive diagnosis algorithms are proposed in [6] and [11]. These algorithms work in two steps. A set of log (n) vectors is generated in the first step. A second set of vectors is generated based on the outcome of the application of the first set of vectors.

III. BIST APPROACH

BIST schemes are developed in this work for the testing of interconnects. The schemes address various issues, including test generation complexity, test vector loading and application, response analysis, and diagnostic resolution. In this work, the boundary-scan architecture is treated as the framework for the PCB level BIST. Standard boundary-scan cells are extended for the capability to generate time-efficient vectors and for response compaction within the boundary-scan chain. This local response compaction reduces shift out time and simplifies detection and diagnosis as well. Detection and diagnosis are made independent of the topology and structural complexity of the nets under test. Detection and diagnosis schemes are proposed in the following.

A. Walking Sequence BIST Scheme

Just as in any other BIST scheme, the walking sequence BIST scheme proposed below is based on the assumption that test generation, test application, and test verification are performed on the PCB under test. In particular, we expect the arrangement in which a test and diagnosis controller (TDC) controls the testability bus and provides the BIST functions. The design of a TDC is proposed in [20].

In the scheme proposed, the test vectors are obtained using a walking one/walking zero sequence. Moreover, simple modifications in the input pin boundary-scan cells are required to obtain a "parity"-based signature for each input pin cell. We will now describe the scheme in detail and prove that the parity signatures provide enough capability for detection and diagnosis of all shorts, stuck-ats and open faults. Moreover, it will be shown that if detection, not diagnosis, is the only desired objective, then the parity signatures can be compacted further into a single count without losing any detectability information. In all the following discussions, we will assume that N, the total number of scan cells on the PCB, is even. If N is odd, TDC can adjust the length of the scan chain to make it even [10].

1) Extension of Boundary Scan Cells

a) Input Pin Scan Cell: Consider the extended scan cell shown in Fig. 5. This scan cell has all the functionality of a standard boundary scan cell. The box marked X shows the additional logic which is not part of the standard [15] cell. Registers $R1$ and $R2$ are used for loading and shifting of the walking sequence. The additional register $R3$, together with the two-input XOR gate, forms the parity checking circuitry for response compaction. With this arrangement, the scan cell can carry out the function of shifting a loaded vector in the boundary scan chain while at the same time compacting the bits coming into the scan cell from the net to which it is connected. The important point here, of course, is that incoming bits on the net do not affect the contents of the shifted sequence in the boundary-scan chain. In other words, these two operations—the test vectors obtained by shifting inside the boundary-scan chain and the response compaction—are carried out independently. When the testing sequence is finished, the compacted "parity" bit at $R3$ can be transferred (see MUX1 in Fig. 5) to the regular boundary-scan register $R1$ and shifted out to the TDC.

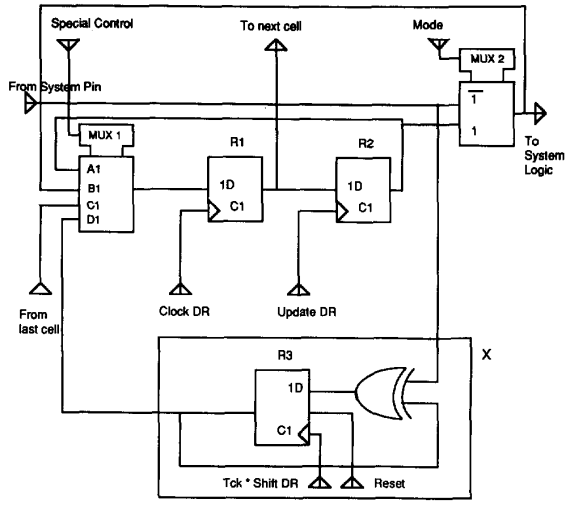


Fig. 5. Extended input pin scan cell.

b) Output Pin Scan Cell: Walking sequence vectors are applied using a shift-update-capture cycle. The output pin scan cell is extended, as shown in Fig. 6, to capture itself during each capture operation of the interconnect test. This guarantees that the walking sequence is not destroyed due to the capture operation and can be used for local generation of vectors.

c) Control Cell: In the proposed BIST scheme, test vectors are generated by internal shifting operations within the boundary-scan chain. On one hand, this requires that the content of each scan cell is changed after each apply-compact-shift cycle. On the other hand, the content of each control cell is to kept fixed during the application of a complete set of test vectors. These two requirements contradict each other. Therefore, internally generated vector sets cannot be implemented using the control cells described in [15]. An extension of this control cell design is proposed in the following.

The extended control cell, shown in Fig. 7, has a bypass capability. The register $R1$, is used for shifting a bitstream along the scan chain. Register $R3$ is loaded with the control bit. At the beginning, the control bits are shifted using $R1$ in each control cell and loaded into $R3$. This loading is performed using proper instruction from the instruction register. After loading the control cells, test vectors are shifted in and loaded in the I/O scan cells. During test vector shifting and loading, $R1$ in each control cell is used to bypass the corresponding control cell. This bypass capability allows the generation of test vectors by internal shifting of the loaded vector, while the contents of the control cells remain fixed. The register $R2$ is used to load the control data when bypass capability is not needed. Finally, the control cell has an extension for non-destructive capture operation similar to that of the extended output pin scan cell.

d) Bidirectional Pin Scan Cell: A bidirectional pin should be configured as an input or an output pin scan cell

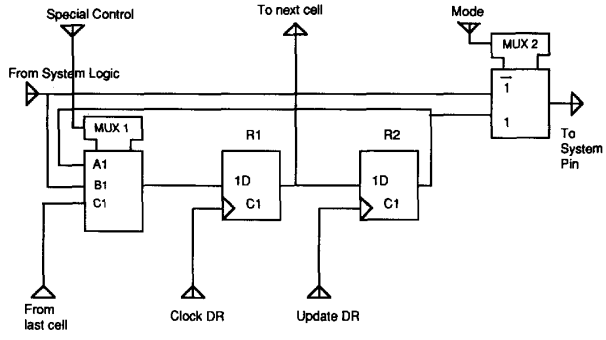


Fig. 6. Extended output pin scan cell.

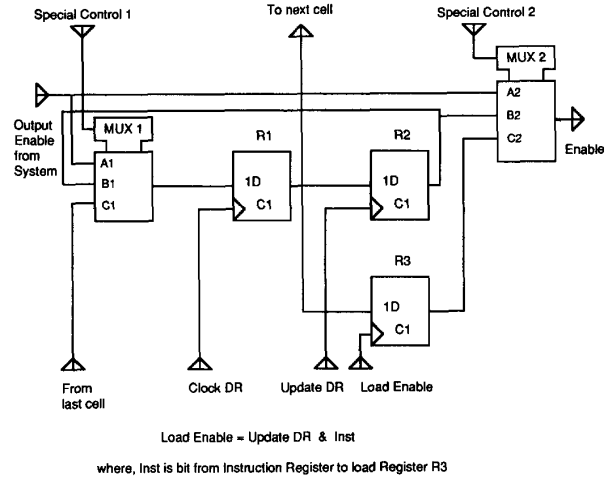


Fig. 7. Extended control cell.

during interconnect test. One of the implementations of bidirectional pin scan cells in [15] consists of a control cell, and input cell and an output cell. These cells of a bidirectional pin would require extensions as described above.

(ii) Test Generation

Consider the N-bit long binary sequence of a parallel test vector (PTV) with a single 1 followed by $(N-1)$ 0's as shown below:

$$1000 \cdots 0.$$

This PTV can be easily generated by the TDC and loaded into the N cell boundary-scan chain in such a way that 1 is in the cell closest to the scan-in edge of the PCB. Assuming that $N(1)$ is the "selected" output pin scan cell of net $n(1)$, if this PTV is applied and the output is collected, then at each input pin scan cell, except those connected to net $n(1)$, the bit arriving into the compaction part is 0. If we now shift this PTV by one cell position, the resulting PTV is "0100 $\cdots$ 0" (assuming TDC feeds 0 in $N(1)$). If the second cell in the boundary-scan chain is the

TABLE II
WALKING SEQUENCES

	Walking One					Walking Zero				
Nets	$V(10)$	$V(9)$	$\cdots$	$V(2)$	$V(1)$	$V(10)$	$V(9)$	$\cdots$	$V(2)$	$V(1)$
$n(1)$	0	0	$\cdots$	0	1	1	1	$\cdots$	1	0
	0	0	$\cdots$	1	0	1	1	$\cdots$	0	1
$n(2)$	0	0	$\cdots$	0	0	1	1	$\cdots$	1	1
$n(3)$	0	0	$\cdots$	0	0	1	1	$\cdots$	1	1
	0	0	$\cdots$	0	0	1	1	$\cdots$	1	1
	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
$n(5)$	0	0	$\cdots$	0	0	1	1	$\cdots$	1	1
	0	1	$\cdots$	0	0	1	0	$\cdots$	1	1
$n(6)$	1	0	$\cdots$	0	0	0	1	$\cdots$	1	1

"selected" (see Section II-A), output pin scan cell of net $n(2)$, then the 1 can be applied to $n(2)$ from this cell. Otherwise, 0 will be applied to every net, and the "1" in the second scan cell will be shifted by one cell position in the next shift instruction to obtain 00100 $\cdots$ 0. Thus, by gradually shifting and applying a walking one sequence, a 1 can be applied to every net, one at a time. Similarly, by shifting and applying a walking zero sequence, i.e., 011 $\cdots$ 1, a 0 can be applied to every net, one at a time. Table II shows a walking one and a walking zero sequence. As stated in Theorem 1, such a sequence forms a complete test set for the network.

Theorem 1: N vectors of a walking one sequence are sufficient to detect every possible single and multiple short and SA fault (except multiple faults on the same net) in a system of n nets.

Proof: It can be seen from any row in Table II that for a walking one sequence, the STV corresponding to every boundary-scan cell consists of a single 1 and $(N-1)$ 0's. It was mentioned in Section II that test vectors are applied from only one selected output pin scan cell of every net. Thus, the STV applied to every net consists of a single 1 and $(N-1)$ 0's. In the case of an SA fault, the number of 1's in the SRV is either increased to N 1's (for SA-1) or decreased to zero 1's (for SA-0).

Notice that every input vector in the walking one sequence contains a single 1. Therefore, 1's cannot be applied to two nets simultaneously. At most, only one net receives a 1 and the remaining nets receive 0's due to each input vector. STVs applied to two nets $n(i)$ and $n(j)$ look like this:

$$n(i) \cdots 00 \cdots 010 \cdots 00 \cdots$$
$$n(j) \cdots 01 \cdots 000 \cdots 00 \cdots .$$

If two such nets are shorted together, the number of 1's in the faulty SRV's is either increased (for an OR short) or decreased (for an AND short). Therefore, any short or SA fault can be detected by applying a walking one sequence. Q.E.D.

An identical proof can be made for a walking zero sequence.

(iii) Response Compaction

As mentioned earlier, output responses are compacted locally within each input pin scan cell. Register $R3$ (Fig. 5) is initialized to 0. The first output bit coming from a net due to the application of the first vector of the walking sequence is XORed with the content of $R3$. The result is stored back in the register $R3$ and XORed with the next bit. This procedure is repeated N times for the N vectors and finally a one-bit compacted response is obtained in $R3$.

For a walking one sequence with N bits (N is even by definition), a single 1 is applied to each net. For a walking zero sequence with N bits, $(N-1)$ 1's are applied to each net. In both cases, the XOR gate is fed an odd number of 1's. Therefore, the fault-free compacted response is 1 for each of the walking sequences.

For a walking one sequence, an SA fault feeds an even number of 1's to the XOR gate (zero 1's for SA-0 and N 1's for SA-1). Also, an OR short among an even number of nets feeds an even number of 1's and any AND short feeds zero 1's to the XOR gate of each faulty net. For all of these faults, the compacted response is 0. However, for an OR-short among an odd number of nets, an odd number of 1's are fed to the XOR gate of each input pin scan cell of each faulty net. Therefore, the compacted response is 1 (the same as the fault-free compacted response). But in this case, the walking zero sequence feeds N 1's to the XOR gate. This makes the compacted response for the walking zero sequence to be 0. Therefore, the fault is detected.

Table III shows the compacted responses obtained by applying both the walking sequences. It can be seen from this table that the fault-free compacted bit-pair (for walking one and zero sequences) is different from any fault compacted bit-pair.

(iv) Detection

For the detection of shorts and SA's, compacted responses in the boundary-scan chain are shifted out and compacted once more in the TDC. The response compactor in this stage is a 1's counter. After applying N vectors of the walking one sequence, the single-bit compacted response from each input pin scan cell is shifted out into the response compactor. The same procedure is repeated for the walking zero sequence.

Theorem 2: Every single and multiple short and SA fault (except multiple faults on the same net) in a system of n nets can be detected by counting the number of 1's shifted

TABLE III
COMPACTED RESPONSES DUE TO WALKING SEQUENCES

	Compacted Response for	
	*W*1	*W*0
No-Fault	1	1
SA-1	0	0
SA-0	0	0
OR Short (Odd)	1	0
OR Short (Even)	0	0
AND Short (Odd)	0	1
AND Short (Even)	0	0

out of the scan chain due to the application of both the walking one and zero sequences and compaction of the responses in each input pin scan cell for each of these sequences.

Proof: Let us suppose there are M input pin scan cells in the boundary scan chain of N scan cells.

Therefore, in the fault-free case, the expected count in the 1's counter

$$CNT = 2M + r$$

where r = (number of 1's in the output pin cells and control cells due to $(N - 1)$ internal shifts of walking one sequence) + (number of 1's in the output pin cells and control scan cells due to $(N - 1)$ internal shifts of walking zero sequence).

Note that a single 1 and $(N - 1)$ 0's are loaded in the scan chain because of the walking one sequence. After $(N - 1)$ internal shifts, the single 1 reaches the last scan cell (i.e., the scan cell closest to the scan out edge of the PCB under test). If this last scan cell is an output pin cell or control cell, the 1 in this cell is shifted out of the scan chain. Each of the remaining output pin cells and control cells contains a 0. Otherwise, if this last scan cell is an input pin scan cell, the 1 in this cell is "destroyed" by the compacted response contained in this cell. In this case, every output pin cell and control cell contains a 0.

For a walking zero sequence, a single 0 and $(N - 1)$ 1's are loaded in the scan chain. The 0 reaches the last scan cell after $(N - 1)$ internal shifts. If this cell is an output pin or control cell, then each of the $(N - M - 1)$ remaining output pin cells and control cells contains a 1. These 1's are shifted out of the scan chain. Else if the last cell is an input pin scan cell, then each of the $(N - M)$ output pin cells and control cells contains a 1.

Therefore, following the argument above, for the last scan cell to be the output pin cell or control cell

$$r = 1 + (N - M - 1)$$
$$= (N - M).$$

For the last scan cell to be input pin cell

$$r = 0 + (N - M)$$
$$= (N - M).$$

Therefore, in both the cases, the total count

$$CNT = 2M + (N - M)$$
$$= (N + M).$$

It can be seen from Table III that the number of 1's obtained from each input pin scan cell of every net is two (one for the walking one sequence and one for the walking zero sequence) in the fault-free case. But for any short or SA fault on any net, as shown in Table III, the number of 1's from each input pin scan cell of the faulty net is reduced to a single 1 or zero 1. In this response compaction scheme, the faults are unidirectional. Every fault reduces the total count. Thus, the faults cannot mask each other.

Therefore, for any type and number of faults, the faulty count:

$$CNT(F) < N + M.$$

Thus,

$$CNT(F) > CNT.$$

Therefore, every short and SA fault is guaranteed to be detected. Q.E.D.

(v) Diagnosis

The walking sequence described above can be used for the diagnosis of shorts and SA's. For diagnosis, however, the second stage of response compaction in the 1's counter is not necessary. From the first stage of response compaction in the input pin scan cells, a pair of compacted response bits corresponding to each of these cells are obtained. One of these bits corresponds to the walking one sequence and the other bit corresponds to the walking zero sequence response compaction. It is shown in Table III that the fault-free compacted bit-pair is 11. However, for any short or SA fault, the response bit-pair is shown to be changed to 01, 10, or 00. Thus, based on the compacted responses shifted out, the net corresponding to each input pin scan cell can be declared faulty or fault-free.

The detection and diagnosis procedures are described in the following, using a pseudocode. The boundary-scan chain is loaded with an N bit vector W1/W0. [W1: 100 $\cdots$ 0 where 1 is loaded in the boundary-scan cell closest to the boundary-scan in edge of the PCB under test and W0: 011, $\cdots$, 1 where 0 is loaded in the boundary-scan cell closest to the boundary-scan in edge of the PCB under test.]. It is assumed that the number of input pin scan cells in the boundary-scan chain is M.

Begin
  Begin
  Repeat for *W*1 and *W*0
    Begin
    Load *W*1/*W*0 in the N bit boundary-scan path

```
        Apply W1/W0
        Collect and compact output at each input pin scan cell
          Begin
          Repeat (N − 1) times
            Begin
            Shift W1/W0 one bit to the right (move a 0/1 in the leftmost cell)
            Apply shifted W1/W0
            Collect and compact response at each input pin scan cell
            End
          End
        Unload the N bit scan path with compacted response
        End
      End
      If detection
        Collect and compact N bit scan path contents in a 1's counter
        Detect based on count in the 1's counter
        If count = N + M Then fault-free
        Else faulty
      Else if diagnosis
        Diagnose based on compacted bit-pair from each input pin scan cell
        If bit-pair = 11 Then fault-free
        Else faulty
    End
```

In the walking sequence scheme, only the first vector of each sequence (walking one and zero) is shifted into the boundary-scan chain. This requires N bits to be shifted in for each sequence. Also, after the response compaction of each sequence, N bits are shifted out for detection. Therefore, the time complexity of this scheme is $O(N)$. Notice that the time complexity of even the minimal-size test set scheme is $O(N \cdot \log n)$ [7].

(vi) Structure Independence and Role of TDC

In the proposed schemes, enable signals of multi-driver nets are to be set properly before the application of walking sequence vectors. This requires structural information regarding connectivity of the nets. The information is used by TDC to set the enable signals properly. Once the setting is done, the generation of walking sequence vectors and the response compaction for detection and diagnosis become structure independent. Test vector generation is independent of the complexity of net structure, order of nets, order of boundary-scan cells, etc. Detection involves counting the number of 1's coming out of the boundary-scan chain after compaction. Diagnosis is local, based on the compacted response of each input pin scan cell. Diagnosis is not dependent on the contents of any other input pin scan cell of the faulty net, or any other net, for that matter. Note that control cells are extended by design and loaded initially to take care of the topology and structural information about the nets. As a result, detection and diagnosis procedures are structure independent.

As mentioned earlier, TDC is assumed to control the testability bus and provide the BIST function. TDC should have the functionality to generate the loading sequence for walking sequence vectors, to generate enable control bits based on structural information provided from outside, to compact data for detection, and to analyze response data for diagnosis. Such a TDC removes the need for a costly automatic test equipment (ATE) or test host. In this case, the test host has only to initiate the test and supply the TDC with structural information regarding enable control cells. However, as an alternative approach it is possible to apply the BIST functions like generating loading sequence, counting compacted bits for detection and analyzing response data for diagnosis from the test host. This latter implementation will require a simpler TDC, mainly to control the testability bus. Therefore, there is a tradeoff between the complexity of the TDC and the cost of the external test host.

B. Testing of Open Faults

Testing of open faults can be divided into two categories. The first category includes the detection of net open faults. In these tests, nets are treated as units under test. The second category involves the diagnosis of open faults. Again, there are two types of diagnosis. The first type, called net open fault diagnosis, treats the nets as the units under test. All the faulty nets are identified in this test. The other type of diagnosis is called net conductor open fault diagnosis. A faulty conductor in the net can be isolated using this approach. Detection and diagnosis of only net open faults are discussed in the following. It is possible to diagnose net conductor open faults by further analyzing compacted responses at different input pin scan cells of a faulty net. Those issues are addressed in [10].

Open faults can be tested by checking for a conducting path from each output pin scan cell to all the input pin scan cells in every net. To do this, input pin scan cells should be initialized to a known logic value. The opposite

logic value should be applied from the output pin scan cell. In the fault-free case, the values in input pin scan cells should be changed through the conducting paths.

It is shown in [10] that the walking sequences satisfy the above condition. Therefore, the walking sequence detection and diagnosis schemes proposed above for shorts and SA's detection and diagnosis can also detect and diagnose open faults in nets.

However, as discussed in Section II, testing of open faults is structure dependent. Thus, a single vector is sufficient for detecting open faults in a simple net (without multiple output pin scan cells). For multiple output pin scan cell nets, vectors should be applied from each output pin scan cell driver one at a time. As defined in Section II-C, there can be at most P output pin scan cells connected to a net. This requires P different settings of the control cells to enable drivers at each output pin scan cell one at a time. These settings will ensure that every conductor in every net is tested for open fault. Consequently, a walking sequence is to be applied P times with P different control settings in order to detect/diagnose every possible open fault in the nets under test.

IV. Conclusion

The problems and complexities of PCB level interconnect testing are addressed in this paper. Boundary-scan architecture [15] is an efficient framework to provide electronic access to the interconnect test points. This paper addresses the interconnect test problem in the context of the boundary-scan architecture. The approach taken here is motivated by the fact that the cost of introducing the design for the testability framework of the boundary-scan architecture can be justified only if the framework is exploited to its full extent for PCB level testing. Therefore, an attempt has been made to use the boundary-scan architecture for every aspect of interconnect testing, from test vector generation to output response analysis. This approach results in BIST schemes for testing of the PCB level interconnects.

Circuitry required for the proposed BIST schemes is incorporated within each boundary-scan cell. Input pin, output pin and control cells described in [15] are extended to perform functions such as local generation of test vectors, local response compaction, etc. These extensions result in very time-efficient BIST schemes. However, this BIST approach requires that every boundary-scan cell of every IC on a PCB should have the proposed modifications in order for the BIST schemes to work. Extended boundary-scan cells required for input pin, output pin, enable control signal, and bidirectional pins are described in the paper. We are presently working towards the optimization of the extended cells with appropriate control signals.

The proposed schemes are developed for a boundary-scan environment. However, it is possible that a number of non-boundary-scan components (glue logic) are used on the PCB. Proposed BIST schemes cannot work in such a partial boundary-scan environment with glue logic. Some modifications of the proposed schemes for partial boundary-scan environment are discussed in [9] and [10].

References

[1] J. Bateson, *In-Circuit Testing*. New York, NY: Van Nostrand Reinhold, 1985.

[2] F. M. Beenker, et al., "Macro testing: unifying IC and board testing," *IEEE Design & Test*, pp. 26–32, Dec. 1986.

[3] R. G. Bennetts, *Introduction to Digital Board Testing*. New York: Crane Russark, 1982.

[4] D. K. Bhavsar, "Testing of interconnections to static RAMS," *IEEE Design & Test*, pp. 63–71, June 1991.

[5] C. S. Gloster Jr. and F. Brglez, "Boundary scan with cellular-based built-in self-test," in *Proc. Int. Test Conf.*, 1988, pp. 138–145.

[6] P. Goel and M. T. McMahon, "Electronic chip-in-place test," in *Proc. Int. Test Conf.*, 1982, pp. 83–90.

[7] M. Gruetzner and H. Grabinski, "Simulation and test of faults in WSI interconnect systems," in *Proc. Int. Conf. on WSI*, 1989, pp. 345–354.

[8] A. Hassan, J. Rajski, and V. K. Agarwal, "Testing and diagnosis of interconnects using boundary scan architecture," in *Proc. Int. Test Conf.*, 1988, pp. 126–137.

[9] A. Hassan, V. K. Agarwal, J. Rajski and B. Nadeau-Dostie, "Testing of glue logic interconnects using boundary scan architecture," in *Proc. of Int. Test Conf.*, 1989, pp. 700–711.

[10] A. Hassan, "Testing of board interconnects using boundary scan architecture," Ph.D. dissertation, Dep. Elect. Eng., McGill Univ., Montreal, Canada, Nov. 1989.

[11] N. Jarwala and C. W. Yau, "A new framework for analyzing test generation and diagnosis algorithms for wiring networks," in *Proc. of Int. Test Conf.*, 1989, pp. 63–70.

[12] F. de Jong, J. S. Matos, and J. M. Ferreira, "Boundary scan test, test methodology and fault modeling," *JETTA*, vol. 2, no. 1, pp. 77–88, Mar. 1991.

[13] W. H. Kautz, "Testing of faults in wiring networks," *IEEE Trans. Computers*, vol. C-23, pp. 358–363, Apr. 1974.

[14] B. Nadeau-Dostie, P. S. Wilcox, and V. K. Agarwal, "A scan-based BIST technique using pair-wise compare of identical components," in *Proc. Fourth CSI/IEEE Int. Symp. VLSI Design*, 1991, pp. 225–230.

[15] *IEEE Standard Test Access Port and Boundary-Scan Architecture*, IEEE Std. 1149.1-1990, 1990.

[16] D. Sterba, A. Halliday, and D. McClean, "ATPG and diagnostics for board interconnect implementing boundary scan," *JETTA*, vol. 2, no. 1, pp. 89–98, Mar. 1991.

[17] *Scan Test Device Octal Buffer*, Texas Instruments, Dallas, TI Data Sheet, 1989.

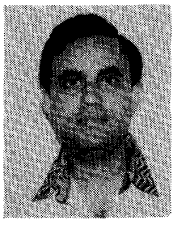
[18] *JTAG & SCOPE Instruction: An Application Brief*, Texas Instruments, Dallas, TI, 1989.

[19] P. T. Wagner, "Interconnect testing with boundary scan," in *Proc. Int. Test Conf.*, 1987, pp. 52–57.

[20] L. T. Wang, M. Marhoefer, and E. J. McCluskey, "A self-test and self-diagnosis architecture for boards using boundary scan," in *Proc. Europ. Test Conf.*, 1989, pp. 119–126.

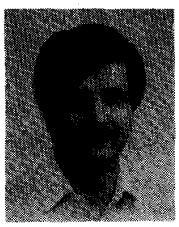
[21] T. W. Williams and K. P. Parker, "Design for testability—A survey," *IEEE Trans. Computers*, vol. C-31, pp. 2–15, Jan. 1982.

[22] C. W. Yau and N. Jarwala, "A unified theory for designing optimal test generation and diagnosis algorithms for board interconnects," in *Proc. Int. Test Conf.*, 1989, pp. 71–77.



Abu S. M. Hassan (S'85–M'89) received the B.Sc. Engg. degree in electrical engineering from Bangladesh University of Engineering and Technology, Dhaka, Bangladesh, in 1982, and the M. Eng. and the Ph.D. degrees in electrical engineering from McGill University, Montreal, Canada, in 1985 and 1989, respectively.

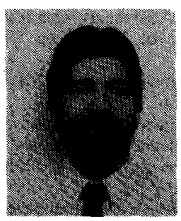
Since 1989 he has been a Member of the Scientific Staff with the Design for Testability Group at Bell Northern Research, Ottawa, Canada. His research interests include boundary-scan based testing, automatic test pattern generation for digital circuits, and built-in self-test.



Vinod K. Agarwal (S'74-M'77-SM'84-F'92) received the B.E. degree in Electronics from Birla Institute of Technology and Science, India, in 1973; the M. Eng. degree in electrical engineering from the University of Pittsburgh in 1975, and the Ph.D. degree in electrical engineering from the Johns Hopkins University, Baltimore, in 1977.

In 1978, he joined the Department of Electrical Engineering of McGill University, Montreal, Canada, where he is currently the Northern Telecom/Bell-Northern Research-National Sciences and Engineering Research Council (NT/BNR-NSERC) Industrial Research Chair Professor for Digital System Design, where he has made significant contributions to design-for-testability strategy at NT/BNR. He has also had consulting affiliations with Kodak, Rochester, NY, Hitachi Central Research Laboratory, Japan, Sunstrand, IL, and Northern Telecom and Bell-Northern Research, Ottawa. He has over 80 publications, including two U.S. patents and various chapters in books on testing and fault-tolerance. His research interests include switching theory, digital testing, software engineering, system level diagnosis, system design methodology, and fault-tolerant computing. He was General Chair of the 1991 IEEE Symposium on Fault-Tolerant Computing, Montreal, and has been on the technical program committees of several IEEE conferences and workshops. He has served as Associate Editor (testing) and guest editor of the IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS. He is a member of the Editorial Boards of the *Journal of Electronic Testing: Theory and Applications* (Kluwer) and *Computer Systems and Engineering* (Butterworth).

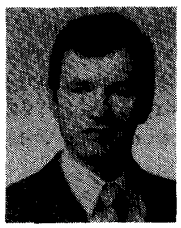
Dr. Agarwal is also a member of IFIP WG. 10.4 on Reliability and Fault-Tolerance, ACM and Sigma Xi.



Benoit Nadeau-Dostie (S'80-M'85) received the Ph.D. degree in electrical engineering from the Université de Sherbrooke, Québec, Canada, in 1985.

He then joined the Department of Electrical Engineering of the Université Laval (Vision and Digital Systems Lab). Since 1986, he has been working for Bell-Northern Research, Ottawa, Canada, on the development of new design-for-testability techniques for communication systems. He has published several papers in the fields of design-for-testability (DFT) and built-in self-test (BIST) including four U.S. patents. He has served on the program committee of several international conferences and workshops.

Dr. Nadeau-Dostie is a member of the Ordre des Ingénieurs du Québec.



Janusz Rajski (A'87) received the M.Eng. degree in electrical engineering from the Technical University of Gdańsk, Poland, in 1973 and the Ph.D. degree in electrical engineering from the Technical University of Poznań, Poland, in 1982.

From 1973 to 1984 he was a member of the faculty of the Technical University of Poznań. Since 1984, he has been with the Department of Electrical Engineering, McGill University, Montreal, Canada, where he is currently an Associate Professor. His main research interests include design automation and testing of VLSI systems, design for testability (DFT), built-in self test (BIST), and logic synthesis. He has published more than 50 research papers in these areas. He has done contract work and has been a consultant in the area of testing to a number of companies. He was a Guest Co-Editor of the June 1990 and January 1992 special issues of the IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS. He is a member of the editorial board of the *Journal of Electronic Testing: Theory and Applications* (Kluwer). He has served on technical program committees of various conferences, coordinated the topic of Automatic Test Pattern Generation for the International Test Conference, and co-chaired the 5th Canadian Workshop on New Directions for Testing in Montreal in May 1992.