Iddq Testing for CMOS VLSI

Rochit Rajsuman, SENIOR MEMBER, IEEE

It is little more than 15-years since the idea of Iddq testing was first proposed. Many semiconductor companies now consider Iddq testing as an integral part of the overall testing for all IC's. This paper describes the present status of Iddq testing along with the essential items and necessary data related to Iddq testing.

As part of the introduction, a historical background and discussion is given on why this test method has drawn attention. A section on physical defects with in-depth discussion and examples is used to illustrate why a test method outside the voltage environment is required. Data with additional information from case studies is used to explain the effectiveness of Iddq testing. In Section IV, design issues, design styles, Iddq test vector generation and simulation methods are discussed. The concern of whether Iddq testing will remain useful in deep submicron technologies is addressed (Section V). The use of Iddq testing for reliability screening is described (Section VI). The current measurement methods for Iddq testing are given (Section VII) followed by comments on the economics of Iddq testing (Section VIII). In Section IX pointers to some recent research are given and finally, concluding remarks are given in Section X.

***Keywords*—*Burn-in, current measurement, current sensor, current testing, deep sub-micron technology, design-for-test, fault diagnosis, fault models, IC testing, Iddq testing, physical defects, reliability screening, reliability testing, semiconductor testing, simulation, system-on-a-chip testing, test economics, test effectiveness, test vectors.*

I. INTRODUCTION

IDDQ testing refers to the integrated circuit (IC) testing method based upon measurement of steady state power-supply current. Iddq stands for quiescent Idd, or quiescent power-supply current. Today, the majority of IC's are manufactured using complementary metal–oxide–semiconductor (CMOS) technology. In steady state, when all switching transients are settled-down, a CMOS circuit dissipates almost zero static current. The leakage current in a defect-free CMOS circuit is negligible (on the order of few nanoamperes). However, in case of a defect such as gate-oxide short or short between two metal lines, a conduction path from power-supply (Vdd) to ground (Gnd) is formed and subsequently the circuit dissipates significantly high current. This faulty current is a few orders of magnitude higher than the fault-free leakage current. Thus, by monitoring the power-supply current, one may distinguish between faulty and fault-free circuits.

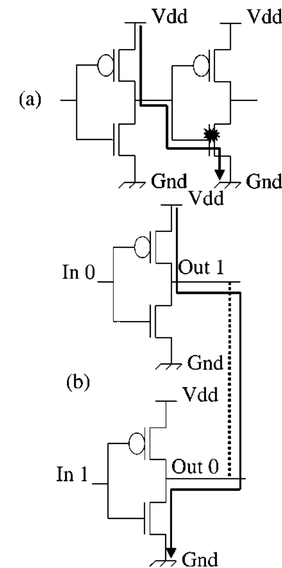

Fig. 1. Formation of current path in presence of gate-oxide short and metal bridging in CMOS circuit.

This concept is illustrated in Fig. 1. Fig. 1(a) shows CMOS inverters with a gate-oxide short and Fig. 1(b) shows two CMOS inverters with shorted outputs. The current conduction paths formed due to these defects are also highlighted in Fig. 1(a) and (b). As suggested by Fig. 1, the concept of Iddq testing is very simple, however, its implementation in today's very large-scale integrated circuits (VLSI) is not so straightforward. This paper covers the present state of this technology and provides necessary details on all essential items. The rest of the introduction section is devoted to the historical background and answering the question why this method has drawn attention and became a buzzword in the semiconductor test industry. These sub-sections essentially provide the motivation to study this technology.

A. Historical Background

Current measurement based testing of electronics components has always been an integral part of the testing since the birth of semiconductor industry. It is used to detect gross shorts and is generally referred to as static Idd test. The present form of quiescent current (Iddq) measurement based testing for CMOS VLSI, known as Iddq testing, was first

Manuscript received August 19, 1998; revised December 13, 1999.
The author is with Advantest America R&D Center, Santa Clara, CA 95054 USA (E-mail: r.rajsuman@advantest-ard.com).
Publisher Item Identifier S 0018-9219(00)02875-9.

0018–9219/00$10.00 © 2000 IEEE

publicly proposed in 1981 [1] and then formulated in [2] and [3] for the detection of bridging faults. Around the same time, researchers at IBM also proposed the monitoring of switching current to detect transient failures (noise related failures) in memory devices [4]. In the following couple of years, a number of labs reported that monitoring quiescent current is an effective method to detect various physical defects such as bridging, gate oxide shorts, inter-gate shorts, stuck-on faults, etc. [5]–[7]. In this early stage, besides government/defense labs (such as Sandia Labs [7]), few commercial semiconductor manufacturers included Iddq testing as part of their overall test program [8]. It is worth mentioning that commercial semiconductor manufacturers have always measured static Idd as part of the parametric test as an integral part of the overall testing. Although, it can be considered as single Iddq measurement, by almost every manufacturer even today, this test is identified by a different name (static Idd test, I-test, easy current test, etc.) and considered separately from Iddq testing, which implies multiple measurements.

By the mid 1980s, semiconductor manufacturers started to recognize Iddq testing as an effective means to detect physical defects. It is worth noticing that long before CMOS became the mainstream, semiconductor companies were aware of the limitations of the stuck-at fault model that many physical defects do not map onto stuck-at faults [9], [10]. Thus, besides having less than 100% stuck-at fault coverage during testing, conventional testing in the voltage environment was not sufficient for higher quality and a testing method targeted toward layout/process oriented defects was needed [9]. Such testing gained acceptance in mid 1990s after Iddq testing was recognized as a cost-effective method.

While most of the work in mid 1980s on current measurement was based upon off-chip measurement circuitry, around 1989 proposals appeared for on-chip current sensors. In the early 1990s, a large number of proposals for current sensors led the IEEE Technical Committee on Test Technology in 1994 to set-up QTAG (Quality Test Action Group) task force to investigate the feasibility of a standard for off-chip current sensor. However, the QTAG task force recognized that the current sensors are not cost effective from chip and equipment manufacturer's point of view and hence, this effort was dropped in 1996. Research on current sensors is now directed toward high-speed off-chip sensors.

In the early 1990s, Iddq testing started to gain acceptance in the commercial semiconductor industry. The defect oriented simulation method such as Inductive Fault Analysis clearly showed why many defects do not map onto stuck-at faults and not detected by the conventional testing [11], [12]. Particularly, as the minimum feature size became less than 1.0 μm, particle defects and bridging became the dominant cause of failure. Since Iddq testing provides physical defect oriented testing, it gained acceptance. Other reasons were the cost-effective testing mechanism requiring little work by the circuit designer, negligible or no area overhead or increase in die-size and a small number of vectors in the Iddq test set. Since late 1980s, many papers started to appear in conferences and journals describing various aspects of Iddq testing. In response, the IEEE Technical Committee on Test Technology approved a new workshop on Iddq testing that was held in-conjunction with International Test Conference 1995.

By the mid–1990s, many companies developed CAD tools for Iddq vectors and a number of EDA and semiconductor companies such as Sunrise (now View Logic), CrossCheck (now Duet), System Science (now Synopsys), Ford Microelectronics, LSI Logic, Lucent, IBM, etc. also commercialized these tools. Some of these tools selected Iddq vectors from a functional test set, while a few tools also included an Iddq ATPG. The fault models used in these tools are stuck-at, pseudo stuck-at, toggle coverage and bridging fault models. Most of the tools work at the gate-level netlist, however, tools such as Power Fault from System Science also work at the RTL netlist and provide early indication if any modification in the design will make it suitable for Iddq testing.

Since Iddq testing is oriented toward physical defects, few people also considered Iddq testing as part of the reliability testing, although many considered it as a supplement to the functional/logic testing. In mid 1990s, a number of studies were conducted to correlate the effectiveness of Iddq testing with conventional reliability screenings (stress testing) and burn-in [13]–[15], [84], [85]. These studies prompted a few companies such as Intel, LSI Logic, etc. to use Iddq testing as a supplement to reliability screening and reduced their standard burn-in time on some products.

In 1996, Semiconductor Research Corporation (SRC) task force identified Iddq and defect oriented testing as one of the key test methodologies with other methodologies such as Core test for the late 1990s and into the 21st century. This task force recommended that the SRC sponsored university research be guided in that direction. The report of this task force has been used widely in industry to fund independent university research. It is expected that sponsored research on various aspects of Iddq testing will continue as well as its use in commercial industry will continue to increase.

B. Reasons for Attention

There are many reasons why Iddq testing has drawn significant attention from test professionals. However, the primary reason is that it is extremely cost effective and uses root cause of problem (physical defect) to identify a bad part. For IC manufacturers, this is an attractive, low cost supplemental test to the functional and stuck-at fault based testing. All the factors in the test cost, such as additional design effort and area, test generation effort, simulation time and test application time, are relatively very small compared to the testing in voltage environment. While increasing the stuck-at fault coverage from 80% to 90%–95% in voltage environment may double the test cost, adding a small Iddq test set is relatively inexpensive and may provide equivalent (sometimes better) benefits. This benefit is qualitatively shown in Fig. 2 [16]. The fault coverage by functional and stuck-at test vectors becomes asymptotic (depending upon circuit and test effort, it is in 90%–99% range), but, after that it requires a large number of additional vectors to get incremental advantage. However, fault coverage can quickly be raised, approaching

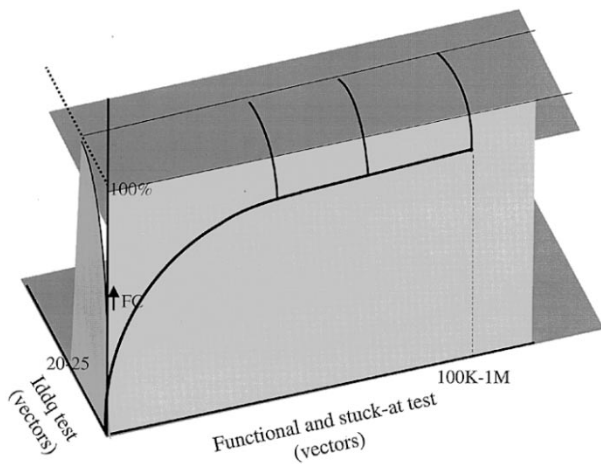

Fig. 2. Fault coverage improvement by adding a small Iddq test.

to 100%, by adding a small Iddq test set to the functional and stuck-at test set. With the available data, it appears that more than 95% fault coverage can be achieved cost-effectively by adding Iddq test set of about 20 vectors to the functional and stuck-at test set with 80%–85% coverage [16]. Increasing fault coverage beyond that requires adding a large number of Iddq vectors or increasing functional and stuck-at coverage by logic testing or both. It should also be noticed that Iddq testing does not check the functionality and hence it is not a replacement of functional testing; it should always be used as a supplemental test.

In the early 1990s, the question that who will pay for the yield loss and additional tester time was debated intensely. However, as time passed, semiconductor manufacturers realized that the benefits of lower customer returns outweigh the cost of additional tester time and perceived yield loss. At the same time, IC users have realized the benefits of higher in-coming quality level and lower system diagnosis. Hence, both semiconductor vendors as well as their customers want to support Iddq testing as part of the production test.

Iddq testing has been shown to shorten time–market. High stuck-at and functional fault coverage is time consuming, but similar quality levels can be obtained by using a small Iddq test set together with a functional test set. The original proof-of-concept designs (with poor functional coverage and no other test mechanism) were required to turn into mass production due to time—market pressure. In such situations, an Iddq test set can provide the necessary quality check without requiring too much re-design or additional test development. By adding as low as 20 Iddq vectors, prototype designs with only 60% functional fault coverage and no other test mechanisms have been turned into volume production [16].

Another advantage of Iddq testing is that it provides massive observability, thus, the test generation effort is very low compared to logic testing. Iddq testing requires only fault sensitization, the fault-effect is observed through the power supply as shown in Fig. 3. Hence, the fault propagation effort during test generation is not needed. It also provides very high detectability capability per Iddq vector.

A further reason is that as the minimum feature size shrinks, many defects of no consequence in large geometry

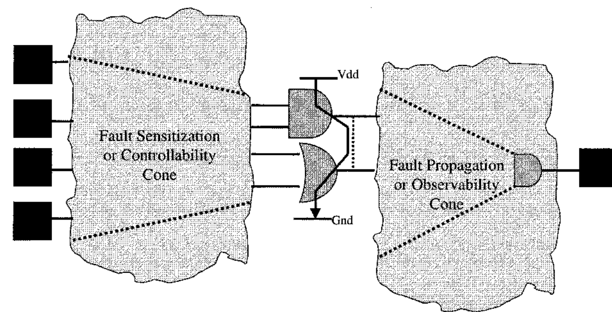

Fig. 3. Fault detection by Iddq testing, only fault sensitization is needed.

become catastrophic in smaller geometry. For example, a particle of size 0.5 μm causing extra metal was relatively unimportant when metal pitch was 2.0 μm, the same defect in a technology with 0.25-μm metal pitch most likely will cause a catastrophic short between metal lines (bridging). Logic testing is quite limited in detection of bridging while Iddq testing is ideally suited for this. There are reports that Iddq testing can also detect defects that do not cause catastrophic failures but only timing related faults (such as resistive bridging and sub-threshold leakage) [93].

Another example is the gate-oxide which is generally grown by dry thermal oxidation between 850 °C to 950 °C. For a 200-Å-thick gate-oxide, ±10 Å variation in the thickness as well as presence of some micro-pores, pin-holes and nonstichiometry oxide at the Si–SiO_2 interface may be acceptable. But the same variation in thickness and quality will result into catastrophic failure in technology with 50–60-Å thick gate-oxide. Conventional logic based testing using the stuck-at fault model is quite limited in detection of these defects. On the other hand, Iddq testing is specifically suitable to detect these and many other process oriented defects. Thus, in some sense, due to the absence of any other testing method to detect process oriented defects, Iddq testing is used.

II. Physical Defects

Since Iddq testing targets physical defects, an overview on physical defects and their testability is given in this section.

In any electrical circuit, opens and shorts are the fundamental physical defects [9], [17]–[20]. Some defects such as partial open and resistive bridging may not cause a gross failure but only timing related error or degraded reliability [21]. Almost all studies on physical defect show that only a small fraction of defects can be modeled at the stuck-at level, the conclusions from these studies are:

1) Wafer defects are found in clusters. These clusters are randomly distributed over the whole wafer. Every part of the wafer has an equal probability of having a defect cluster.
2) Any part of a diffusion, polysilicon, or metal line may have an open fault. Any contact between any two layers may be open.
3) A bridging may occur between any two electrical nodes, whether they belong to one layer or different

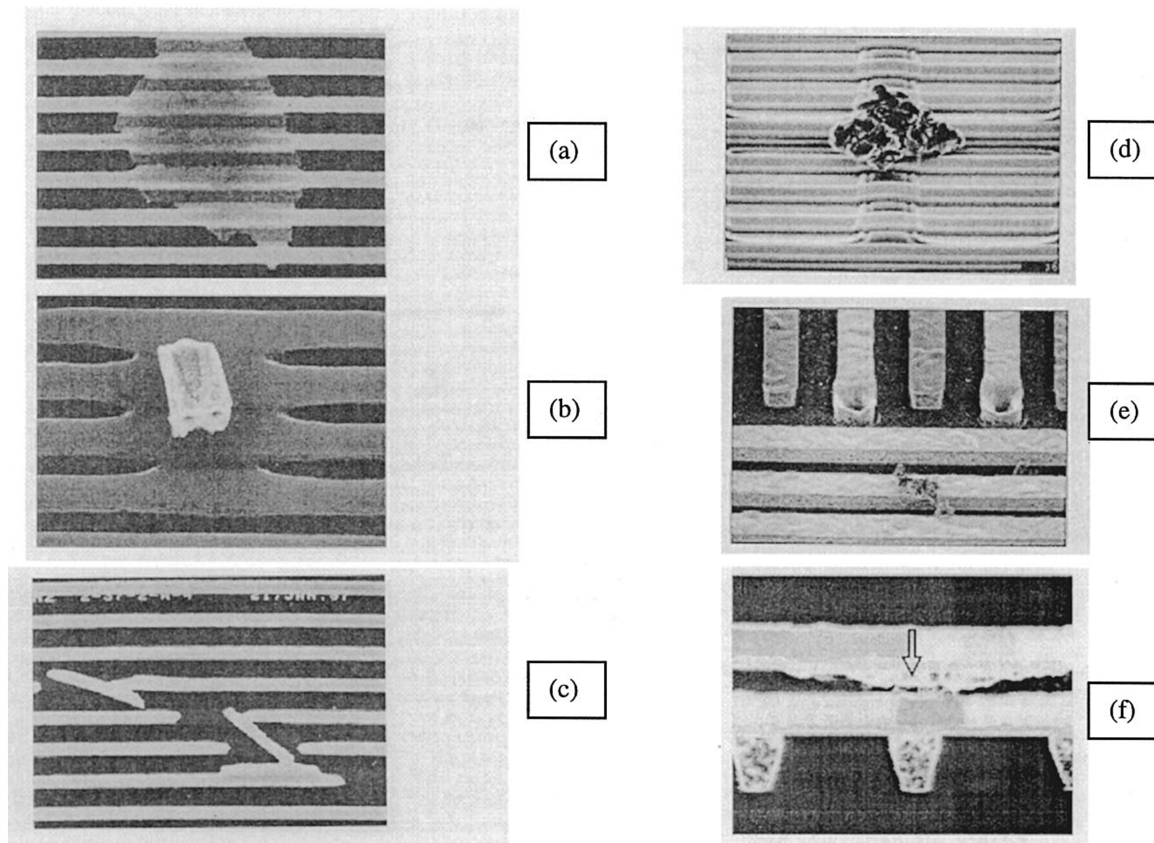

Fig. 4. Example of bridging defects: (a) shorting of seven metal lines caused by unexposed photoresist [38]; (b) shorting of four metal lines by a solid-state particle on the metal mask [38]; (c) shorts and breaks of metal lines caused by a scratch in the photoresist [38]; (d) short among multiple metal lines by a metallization defect of 1 μm in size [R&D magazine, Aug. 1994]; (e) short between two Al lines due to metallization defect [R&D magazine, Aug 1994]; (f) Inter-layer short between two Al interconnects in 0.5 μm technology [40].

layers. Bridging among multiple nodes is equally likely.

4) Only a small percentage of bridging and open faults can be modeled at the stuck-at level. The actual distribution varies and largely depends on the technology and fabrication process.

To understand the effectiveness of Iddq testing, bridging and open defects will be discussed in separate subsections.

A. Bridging (Shorts)

Some examples of bridging defects are shown in Fig. 4. In Fig. 4(a), seven metal lines are bridged together due to unexposed photoresist; in Fig. 4(b), four metal lines are bridged together due to the presence of a foreign particle; in Fig. 4(c), few lines have bridging and opens due to a scratch on the mask; in Fig. 4(d), a 1 μm size killer defect causes catastrophic short; in Fig. 4(e), metallization defect causes a single bridging between two Al lines; and in Fig. 4(f) an inter-layer short is shown. The cause of defects in each of these examples is different, but the results are either bridging or open.

The simulation and modeling of such defects can be done by Inductive Fault Analysis; examples are given in Fig. 5. Circuit schematics are also included in Fig. 5 to illustrate the effect of these defects. Figure 5(a), shows a spot defect causing extra polysilicon. The corresponding transistor level and gate-level schematics show that this defect can be modeled at the stuck-at level. However, the examples of spot defects in Fig. 5(b), (c), and (d) are not modeled at the stuck-at level. In Fig. 5(b), an extra transistor is formed; in Fig. 5(c), two inverters are turned into a NAND gate; and in Fig. 5(d), circuit topology is changed which results in a different Boolean output.

Many papers are available on bridging faults that attempt to detect bridging by logic testing in voltage environment [22]–[26]. In [27] and [28], analytical models were developed to explain the bridging behavior.

The behavior of bridging can be explained by the potential divider rule as shown in Fig. 6(a). The outputs of two logic elements are indicated by subscripts 1 and 2, and bridge resistance is shown as x. Let the resistance that connects the $L(H)$ node to ground (Vdd) be $r_L(r_H)$. If the worst-case (min) output H voltage is H_w, the H-level noise margin is N_H, the worst-case (max) output L voltage is L_w, the L-level noise margin is N_L, and the switching threshold voltage is S_T (the definitions are shown in Fig. 6(b)). The voltage at the output nodes V_1 and V_2 can be given as $V_1 = \text{Vdd}[r_{L1}/(r_{L1}+x+r_{H2})]$ and $V_2 = \text{Vdd}[(r_{L1}+x)/(r_{L1}+x+r_{H2})]$.

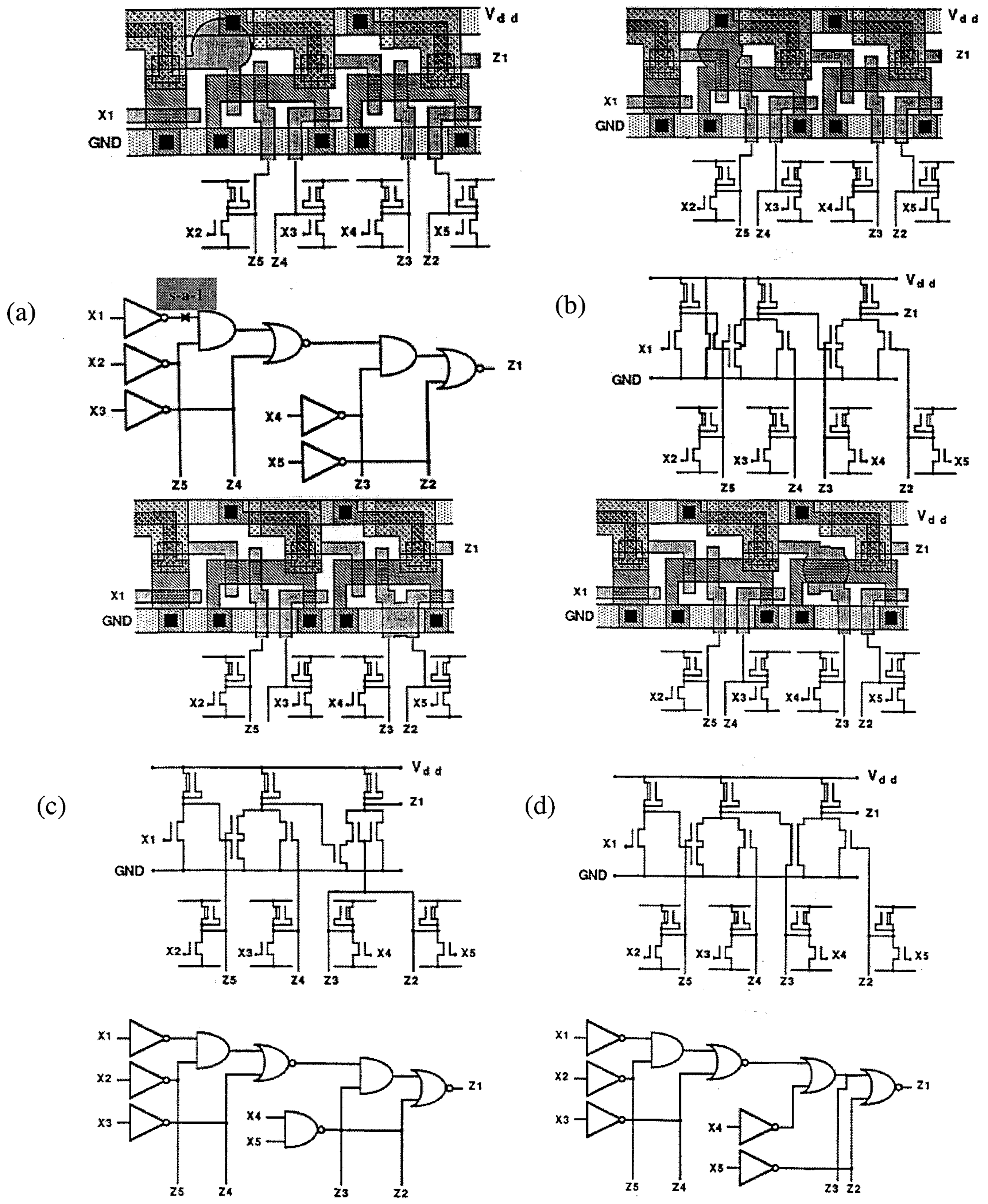

Fig. 5. Examples of spot defects and their effect on the circuit: (a) Spot defect causing extra polysilicon results in a s-a-1 fault [38]; (b) Spot defect causing extra active region results in an extra transistor and a short to Vdd line [38]; (c) Spot defect causing extra polysilicon results in the bridging of inverter outputs consequently transforming them into a NAND gate [38]; (d) Spot defect causing extra polysilicon results in transistor bridging that changes circuit topology and boolean output [38].

Bridging is not consequential if $x \to \infty$, giving $V_1 = 0$ and $V_2 = \mathrm{Vdd}$. For a low-resistance bridging (hard short), $x \to 0$, which gives $V_1 = V_2 = 1/2\mathrm{Vdd}$, when r_L and r_H are equivalent. This potential divider model indicates that in case of high resistance bridging the voltage at the defect-site may or may not cause a logical error depending on the bridge resistance. A hard-short or low resistance bridging will cause $1/2\mathrm{Vdd}$ at the defect-site. The $1/2\mathrm{Vdd}$ implies indeterminate logic value at the defect-site. Thus, even if it is sensitized by logic testing, fault detection is not likely because the indeterminate logic value at the defect-site will become either logic 1 or 0 during fault propagation.

Bridging resistance is important in defect detection. The majority of bridging defects show low resistance, with only about 20% showing significant resistance [29], [30]. Many high resistance bridging defects do not result in failure at the time of testing, and only affect the noise margin (degraded H_W and L_W may not necessarily cause a logical error). However, these defects significantly degrade the device reliability.

Fortunately, a large number of bridging defects (including a large number of the resistive bridging type) can be detected by Iddq testing. The effectiveness of Iddq testing can be visualized by the potential divider rule and by the diagram shown in Fig. 1. As is clear from Fig. 1, in the presence of bridging, a conduction path is formed from Vdd to Gnd. Subsequently, the circuit dissipates a large current through this path, and thus, simple monitoring of the supply current can detect bridging. This detection does not require any fault propagation; the fault propagation is automatic through the power supply.

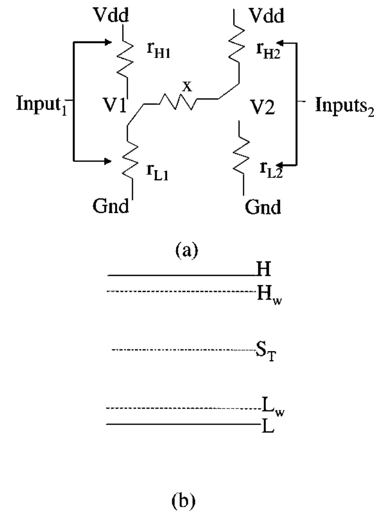

Fig. 6. (a) Simple potential divider model for bridging; (b) Definition of voltage levels [27].

B. Gate Oxide Defects

The various defects and reliability issues in gate-oxide are long known. These defects include pinholes and micropores, dendrites, trapped charge due to hot-carriers, nonstichiometric Si–SiO_2 interface and direct short to diffusion. The examples of gate-oxide short to N^+ diffusion and gate-oxide pinhole are given in Fig. 7. Some of these defects occur during the oxidation or other thermal processes, while other defects may occur due to electrostatic discharge or overstress (ESD/EOS).

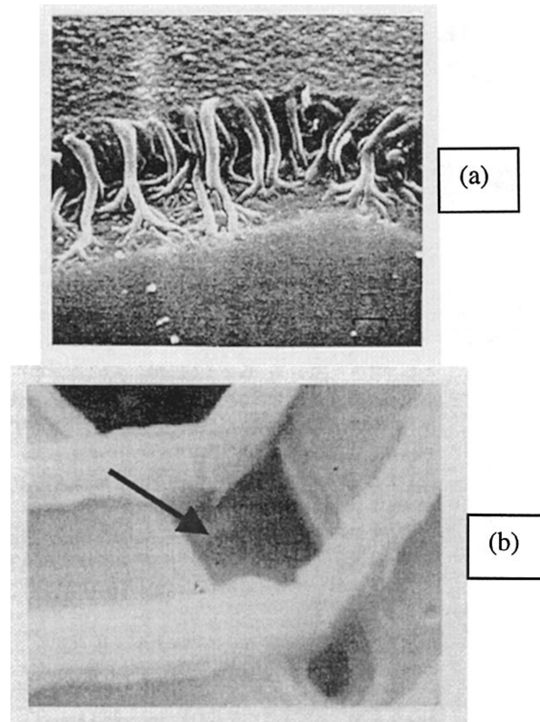

Fig. 7. Examples of gate-oxide defects: (a) Gate-oxide short to N^+ diffusion [31]; (b) Gate-oxide pin-hole causing cell–wordline leakage in a memory [61].

In today's 0.25-μm technology (and below), gate-oxide of 50–60 Å thickness is used for logic MOSFET's and as low as 35–40 Å for devices such as EEPROMs/flash memories. Although, gate-oxide thickness is tightly controlled (most fabs target ±2–3 Å), a smallest variation in thickness increases the possibility of defect. For example, the lower thickness region may cause Fowler–Nordhiem tunneling and in the extreme case, avalanche breakdown during voltage stress test. ESD/EOS induced breakdowns are also very common in such thin oxides.

Gate oxide reliability issues and breakdown mechanisms have been very well reported in the literature. In the majority of cases, gate-oxide defects cause reliability degradation such as change in transistor threshold voltage (V_t) and increased switching delay, only in some cases (such as avalanche breakdown and subsequent short) it causes a logical failure. Some papers also present very elaborate models for gate oxide shorts and defects [32]–[34]. However, in general, logic testing does not detect gate oxide defects [35], [36], primarily due to difficulty in fault-effect propagation. Iddq testing, on the other hand, is very effective in detecting these defects as they cause high current dissipation in the circuit.

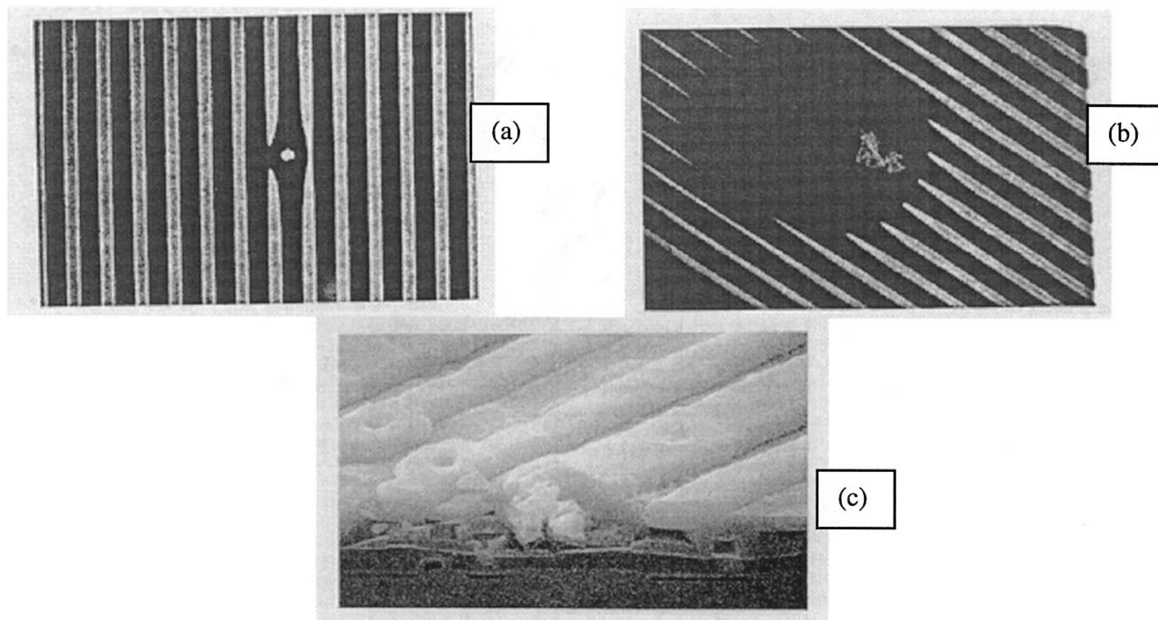

Fig. 8. Examples of open defects: (a) A foreign particle causing a line open and a line thinning [38]; (b) A contaminating particle causing 7-line opens [38]; (c) SEM picture of a defect which caused open in metal 2 and a short in metal 1 [62].

C. Open (Breaks)

Analysis of physical defects from fabs as well as Inductive Fault Analysis suggests that approximately 40% defects are open [37]. Open defects are much more difficult to detect by logic testing. Also, Iddq testing does not necessarily detect them. Examples of open defects include line open, line thinning (it may or may not be a partial open at the time of testing), resistive vias, open vias, etc. A few examples are given in Fig. 3(c) and in Fig. 8(a) and (b) [38]. Figure 8(c), shows yet another example of a defect that caused both open and short [62].

In CMOS circuits, many opens cause sequential behavior. The difficulty in detection of open fault can be illustrated by a simple example. Figure 9 shows a two input NOR gate with an open in drain–output connection of nMOS with input B. All four test vectors are shown in the top part of the adjacent table. As marked in the table, the vector $AB = 01$ is the only vector which sensitizes this open. However, during this vector, in presence of open, output of the gate is in high impedance. Hence, the vector before the sensitization vector defines the logic value at the output. If it is $AB = 10$ or 11, the output remains at 0 and open is not detected. To detect this open, the necessary sequence of patterns is $AB = 00, 01$.

A large number of papers have been written regarding the detection of opens by logic testing using two or multipattern tests [39]–[41]. There are also a large number of reports showing that due to difference in delays along various paths and charge sharing among internal nodes of a gate, two or multipattern tests may become invalidated, these papers also suggest to use testable designs [41]–[44]. While another set of papers suggests to use robust two or multipattern test sequences [45], [46]. Besides causing logical failures, open defects can also cause timing related errors; particularly open-gate defects are very sensitive to capacitive coupling. Thus, suggestions have been made to detect open defects using two or multipattern tests developed to detect delay faults [47], [48].

Both testable designs and two or multipattern sequences (including robust sequences) do not provide a practical solution to detect open defects. Testable designs require redesign of the standard cell library and massive routing of additional global signals, and robust test sequences require elimination of all possible glitches in the circuit and thus, these are difficult to generate. While few companies have developed internal tools to generate two pattern tests for opens, so far, neither universities nor EDA industry has been successful in developing a suitable ATPG tool and pattern sequencing mechanism to deterministically detect open defects in a cost-effective manner by logic testing.

Unfortunately, Iddq testing is also not very effective for open defects. Although, there are some reports that suggest Iddq testing can be used to detect opens [48], [49], such detection is highly subjective to the cell design style and the topology of the circuit. This behavior is clearly explained by a detailed electrical model of open, such as given in [50]. In [51], the effectiveness of Iddq testing was evaluated for open defects by intentionally fabricating an open defect, and it was concluded that Iddq testing is not very effective. In a simple example such as given in Fig. 8, there is no current dissipation when the two pattern test (such as $AB = 01, 10$) is used which also fails to detect open defects in logic testing. Even when the two pattern test (such as $AB = 11, 10$) is used which detects open in logic testing, there is no static current dissipation in the circuit. Hence, in both situations, Iddq testing remains ineffective to detect this open.

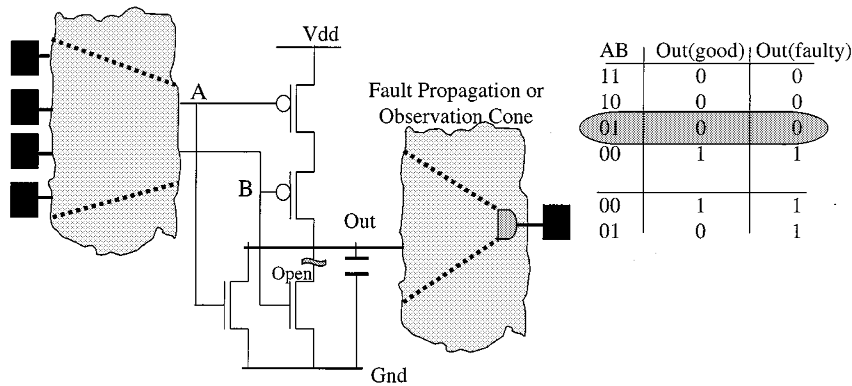

Fig. 9. Example to illustrate difficulty in detection of an open fault. During $AB = 01$, the node marked Out is in high impedance state but from the chip's pin, it still appears at "0." The two pattern sequence $AB = 00, 01$ detects this fault.

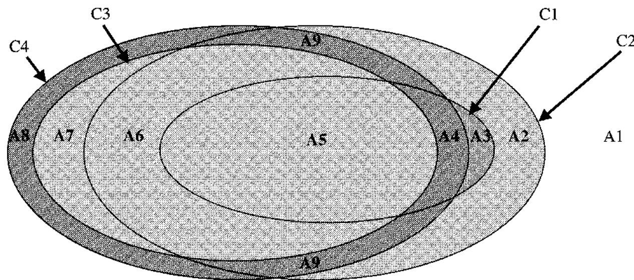

Fig. 10. Relationship in the effectiveness of Iddq and logic testing. A1 are faults not detected by either Iddq or logic testing; C1 faults are detected by logic testing; C2 are all logic faults; C3 faults are detected by Iddq testing; C4 are all Iddq testable faults [55].

Scan based stuck-at	Scan based delay	Iddq: Pass / Functional: Pass	Iddq: Pass / Functional: Fail	Iddq: Fail / Functional: Pass	Iddq: Fail / Functional: Fail
Pass	Pass		6	1463	7
Pass	Fail	14	0	34	1
Fail	Pass	6	1	13	8
Fail	Fail	52	36	1251	

Fig. 11. Summary of data from SEMATECH study [59].

III. Effectiveness of Iddq Testing

With the background of Section II, it is clear that Iddq testing is very effective for defects such as bridging and gate-oxide shorts, but not so effective for opens. Since it does not verify the functionality, it is used as a supplemental test to functional testing. The same is also true about its use for reliability testing such as stress testing and Burn-in. In both functional and reliability testing, Iddq testing adds significant value by improving the quality of the test at a very low cost. Since its infancy, studies have shown that a large percentage of defects can be detected by Iddq testing, however, in almost all studies, there are also failures identified by either functional or scan based testing which remained undetected during Iddq testing [52]–[54]. This relationship is qualitatively shown in Fig. 10 [55].

In Fig. 10, (A2+A9+A6) are undetected faults in logic testing. (A2+A3) are non-Iddq testable faults. A3 are non-Iddq testable faults that are detected by logic testing. (A4+A9+A8) are undetected faults in Iddq testing. A5 are the faults detected by both logic and Iddq testing. A6 are faults undetected in logic testing but detected by Iddq testing. A7 are nonlogical faults that are detected by Iddq testing; and A9 are fault undetected by both logic and Iddq testing.

The net benefit of Iddq testing is (A6+A7). Even for the Iddq test set of just 10–100 vectors, (C4−C3) is generally small. For a design when (C2−C1) is large (for example, functional coverage is only 60%), the overall coverage (C1 ∪ C3) can still be made sufficiently high to be acceptable without incurring significant test development cost.

This conclusion was also drawn by experimental studies to evaluate the effectiveness of various test techniques [56]–[60]. In all studies, Iddq testing detected majority of defects, however, there were few defects, which were undetected by Iddq testing but detected by logic testing. At the same time, some defects remain undetected by logic testing but detected by the Iddq testing. A summary of data from SEMATECH study is given in Fig. 11 [59].

The data in Fig. 11 is a typical representation of studies of this nature, which compare the effectiveness of various testing methods. As marked in Fig. 11, there are 36 devices that passed Iddq test but failed every other test. On the other hand, as marked on the first row, there are 1463 devices failed Iddq test (Iddq threshold limit 5 μA), but passed every other test. This particular study was based on a 116 K-gates standard cell graphics controller chip designed in IBM Phoenix CMOS4LP technology. This technology has 0.45 μm L_{eff} and 0.8 μm L_{drawn}. According to [102], "the sample size was 20 000 devices through wafer test of which

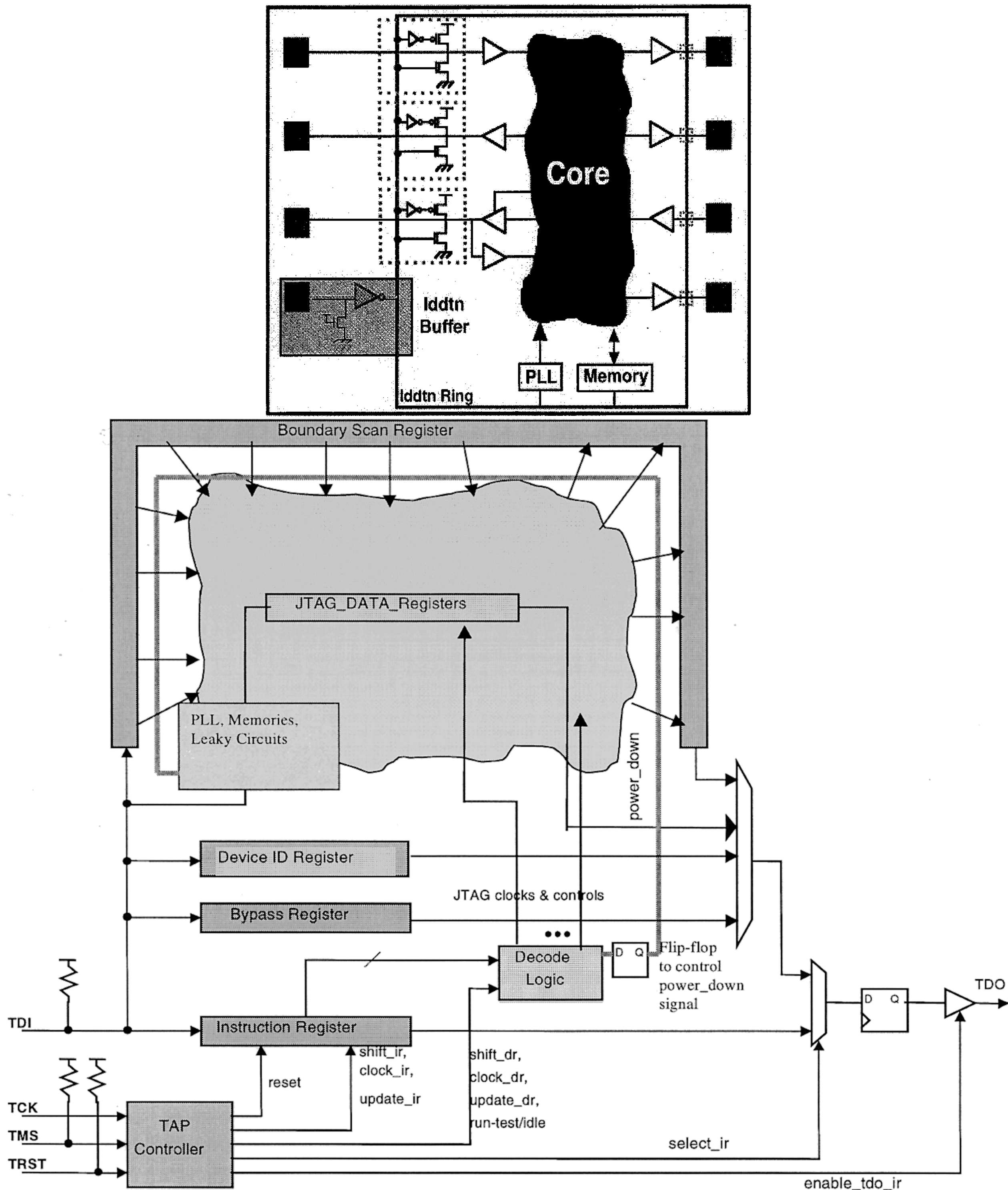

Fig. 12. The AIS design features to facilitate Iddq testing: (a) Iddtn-Ring and Iddtn-Buffer to switch-off static current dissipating logic [16]; (b) Boundary scan TAP controller based design to control static current dissipating logic.

more than 4000 were carefully selected for more complete package level testing and analysis." The chip characteristics were [102]:

- 249 signals I/Os, flip-chip/C4 wafer contacts, 304-pin C4FP package
- Full-scan design, 5280 LSSD latches; boundary scan
- Functional speed 40/50 MHz, function being bus interface controller
- Designed for Iddq testing, typical Iddq $< 1\ \mu$A
- Scan based stuck-at test set with 99.7% stuck-at fault coverage; Functional test set with design verification vectors of 52% coverage; Scan based delay testing with

>90% transition fault coverage; Iddq test set with 195 vectors.

All packaged devices also experienced a minimum of 6 hours of burn-in, while a large sample had 150 hours of burn-in. Identical tests were used at wafer-sort, pre and post burn-in package level test.

Detailed failure analysis was done to identify the defects on a number of devices that failed one or more tests. It was found that Iddq testing detected various kind of bridging (not detected by any other test), including low and high resistive bridging (75 Ω, 194 Ω, 1.63 kΩ, 184 kΩ, 194 kΩ, 340 kΩ). However, an open resulting in a floating gate was not detected by Iddq testing, it was detected by scan based stuck-at testing.

Based upon the above discussion and various studies, the effectiveness of Iddq testing can be summarized as: *Iddq testing is not a panacea, it is a high quality supplemental test that can improve the overall fault coverage significantly without incurring significant test development cost.*

IV. Design-for-Iddq Testing and Test Vector Generation

This section provides some basic rules that are helpful for Iddq testing. The later part of this section describes the Iddq vector generation tools and methods.

A. Design-for-Iddqability

In [16], Design-for-Iddqability has been defined as the incorporation of features that can help in obtaining circuit's quiescent states and design styles that avoid high static current states in the circuit. A number of design rules have been identified to make a design suitable for Iddq testing [16], [63], [64]. The basic philosophy behind these rules is to avoid any possible static high current state in the circuit; if a high current state is unavoidable, then re-design so that it can be isolated during Iddq testing. These rules can be summarized as follows:

1) The circuit should be properly initialized; all flip-flops (registers) should be in a known state. This initialization can be done by a set/reset signal or through scan operation.
2) All static current dissipating logic should be switched off, this includes memory sense-amps, dynamic logic, asynchronous logic, pull-up/pull-down resistors, special I/O buffers and analog circuitry.
3) The circuit should be stable at the strobe point; there should be no pending events.
4) All inputs and bi-directional pins should be either at 0 or at 1.
5) If an input, output or bi-directional pin is pulled-up, it should be at logic 1 connected to Vdd through an on pMOS; if pulled down then it should be at logic 0 connected to Gnd through an on nMOS.
6) All primitive nets with single driver should be checked for the following: a) all nets are either at logic 0 or at logic 1; b) if a net is at x, either the driver should not be tri-stateable or driven by a tri-stateable gate whose enable pin is active; c) any net should not be at z. These conditions ensure that there is no internal bus conflict or floating nodes.
7) Primitive nets driven by multiple drivers should be checked for: a) the net should not be driven to both 1 and 0 simultaneously; b) the net should not be driven simultaneously by multiple drivers to 0 and $x; x$ and $x; 0, 0$ and $x; 1$ and x, in all these conditions there is a potential conflict on net; c) the net should not be driven simultaneously by multiple drivers to x and $z; z, z$ and $x; z, x$ and x, in these situations the net is potentially floating.
8) All nets should be checked so that there is no weak value feeding to a gate during Iddq measurement. Similarly, there should not be a degraded logic value on a node feeding to a gate during Iddq measurement.
9) Special circuit structures should be avoided as much as possible. When such structures are unavoidable, a mechanism should be provided to switch-off these structures during Iddq testing. The examples of such structures are gate and drain/source of a transistor be driven by the same transistor group; feedback and control loops within one transistor group; substrate connection of the transistor should not be floating.
10) A standard cell library which contains components with low power switches and uses a separate power supply for digital logic, I/O pad ring and analog circuit is also helpful. In this situation, Iddq testing on digital logic can be done easily.

Full circuit initialization is a fundamental requirement of Iddq testing. Besides using set/reset of flip-flops or a dedicated signal, full-scan or partial scan can also be used very effectively to initialize the circuit [16], [65].

A number of elegant design methods to isolate static current dissipating logic have been reported. Examples of such designs are given in Fig. 12. Companies such as LSI Logic have adopted design features for global control on power supply (Iddtn-Ring) as an essential component in their design flow and automated it by designing standard cell libraries with Iddtn-signal and Iddtn-Buffer as shown in Fig. 12(a) [16], [66]. Assigning a dedicated pin to facilitate Iddq testing can be viewed as a costly proposition, controlling the static current dissipating logic through Boundary Scan TAP controller is more popular [110]. A simple private JTAG Boundary Scan instruction is sufficient to drive a global signal which switches-off static current dissipating logic as shown in Fig. 12(b). It is also worthwhile to notice that if individual components (analog circuits, memory sense-amp, pull-up/pull-down, dynamic logic, etc.) are designed with a power-down control signal and the name of this signal is consistent throughout the design, then no additional design effort is needed for global control signal. The router sees the same name throughout the chip and connects them into one global signal.

The global power-down control signal based design methodologies are also very important for system-on-a-chip (SoC) designs using embedded cores. SoC designs have millions of transistors and need some kind of partitioning

method for Iddq testing. Generally, these designs also contain multiple power supplies (3.3 V, 2.5 V, 1.8 V, etc.); hence, Iddq testing is performed on one power supply at a time. When multiple embedded cores use the same power supply, resolution of Iddq testing becomes quite low. The power-down control signal based methodology allows selective switch-off of embedded cores and Iddq testing on individual cores, one core at a time [110]. If the implementation is through the TAP controller, one power-down instruction per core can be implemented to obtain independent control for that core during Iddq testing (Fig. 13).

The method shown in Figs. 12 and 13 is also applicable to the IEEE P1500 standard currently under development [111]. Instead of an extra register in the boundary scan for controlling the power-down signals, an equivalent register can be implemented in the Control Scan Path (CSP) of the P1500 with exactly the same functionality as given in Figs. 12 and 13.

B. Iddq Test Vector Generation

A number of tools have been developed internally by corporate CAD and EDA companies as well as by the universities. While most of the tools select vectors from the functional test set based upon the user's defined constraints, some tools also contain an Iddq ATPG. The general characteristics of these tools are the following.

1) Obtain Iddq vectors for the stuck-at coverage or toggle coverage. The input for this option is usually the design netlist and testbench. The vectors are selected from the testbench (Verilog or VHDL) vectors.
2) Obtain Iddq vectors for pseudo stuck-at (PSA) coverage (PSA fault model is similar to the stuck-at fault model. However, Iddq oriented fault simulation does not require fault-effect propagation through the whole circuit; it is propagated only through one gate and observed through the power-supply. Thus, the coverage numbers in such a simulation are considered to be as pseudo stuck-at [52], [97]). The input for this option is the gate-level netlist and testbench.
3) Obtain Iddq vectors for toggle coverage. The input for this option is generally the gate-level netlist.
4) Generate Iddq vectors for the bridging fault model. The input for this option is generally gate-level netlist and testbench. This option is generally associated with an Iddq ATPG, which generates tests by providing opposite logic values on two lines (one line being 1 and another being 0).
5) Generate Iddq vectors for physical defects. The input for this option is the layout (GDSII) and $\min / \max$ particle size (defined by the user). This option is associated with an Iddq ATPG. The tools using this option are experimental; commercial tools are not yet available with this option.

There are also various simulation and timing requirements for a tool to correctly select/generate Iddq vectors and the fault coverage report. Generally, an Iddq Test tool (ATPG or

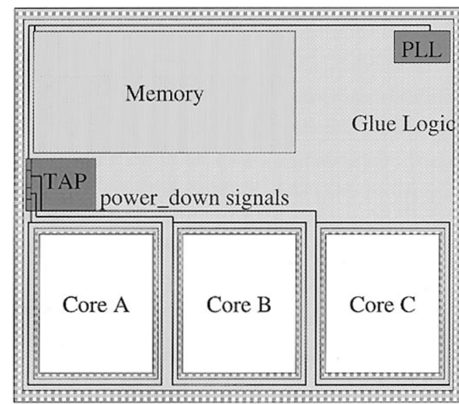

Fig. 13. Implementation of power_down control signals to perform Iddq testing on embedded cores based system-on-a-chip.

vector selection tool) is linked with a Verilog/VHDL simulator. This link is established before the design simulation run. The user's defined constraints for Iddq vectors are passed through a simulation control file that is similar to configuration file of Verilog simulation. The constraints and link to design simulation let the tool identify any Iddq rule violation. A user can waive any of the violations after reviewing them. However, because of these violations, many vectors are dropped from being candidates in the Iddq test set (ideally, if there is no violation, all functional vectors can become Iddq vectors). The remaining vectors are "qualified Iddq vectors," these vectors are fault graded under a fault model specified by the user. A table is created which lists the vectors as well as detected faults; the format of this table varies for various tools. Based upon user's specified constraints the necessary vectors are taken from this table to form the Iddq test set. For example, the Iddq test set may be 25 best vectors or n-vectors that provide 90% coverage, etc. This Iddq test set with a fault coverage number is reported in a separate file.

The vector selection process is generally based upon one of the two procedures:

Procedure Vector Selection (A):

- Step 1: Select a test vector that detects the maximum number of outstanding faults under the user's selected fault model. Add this vector into the list of selected vectors.
- Step 2: Remove all faults detected by the selected vector from consideration.
- Step 3: If the number of selected test vectors exceeds the user's defined limit, exit.
- Step 4: Repeat steps 1–3.
- Step 5: Provide fault coverage of selected vectors.

Procedure Vector Selection (B):

- Step 1: Count the number of test vectors which detect each fault.
- Step 2: Select all faults that are detected by the minimum number of test vectors. Mark the test vectors that detect these faults.
- Step 3: Choose a test vector from the set of test vectors in step 2, which detects the maximum number of uncaught faults.

- Step 4: Repeat steps 1–3 until all faults are detected.

The vector generation process, targeted for bridging faults is generally as follows:

Procedure Vector Generation for Bridging:

- Step 1: Let nodes j and k be the electrical nodes on two ends of the fault. Find the set N_j^0 of input vectors which cause node j to be 0. Find the set N_j^1 of input vectors which causes node j to be 1.
- Step 2: Repeat step 1 for node k.
- Step 3: Compute $T_i = (N_j^0 \cap N_k^1) \cup (N_j^1 \cap N_k^0)$.
- Step 4: Use either procedure A or B above to select necessary vectors from step 3.

Few tools list undetected faults in a separate file and also create a file of strobe time for each vector. Cycle splitting has also been used to catch faults during strobes when the clock is high and again when the clock is low. However, this splitting is possible only for return–zero (RZ) and return–one (RTO) clock formats used in ATE (automatic test equipment) test programs. Generally, one strobe at the end of the cycle is sufficient if only one clock edge latches the data in the storage elements and the other clock edge does not cause any node to switch state. Test patterns for bi-directional pins are such that they change only on the test cycle boundary. When bi-directional pins change state, there is no strobe on one cycle after the enable pin switches its state. From a user's perspective, a generalized overall flow of an Iddq Test tool with necessary file structure is similar to as shown in Fig. 14. The format of files varies from one tool to another. Also, some tools may not provide all the files as shown in Fig. 14. The operational mode of tools also varies from interactive mode to batch mode.

A number of commercial tools from EDA companies Sunrise, Mentor Graphics, System Science, CrossCheck, Syntest, etc. and from IC manufacturers Ford Microelectronics, IBM, Lucent, LSI Logic, Philips Microelectronics, Texas Instruments, NEC Microelectronics, etc. are available to develop Iddq test sets. Some of the tools from IC manufacturers can also be licensed, such as Quietest from Ford Microelectronics, Iddalyzer from LSI Logic, GenTest from Lucent, and Testbench from IBM. Some tools of similar capabilities have also been developed at universities [67]–[71].

Although, there is debate on how many vectors should be used and which are the best vectors. A number of studies, including [52] and the Sematech experiment S-121 indicate that the best benefit is obtained by about 20 vectors obtained for the highest bridging coverage (fault graded under bridging model). It should be noted that in most cases these 20 vectors will not provide 100% coverage under any assumed fault model, it is only a suggested cut-off point on cost-coverage trade-off curve above which sufficiently more vectors will be required for higher fault coverage. Figure 15 shows the nature of coverage by various Iddq test sets.

The fault models used in Iddq test generators are (i) Stuck-at, (ii) pseudo stuck-at, (iii) toggle coverage, (iv) bridging, and (v) defects (in experimental tools). The closest defect model based tools are Ford Microelectronics Quietest [72] and CrossCheck's (now Duet) CM-I tools. These tools use a special fault mode for each cell type in which each

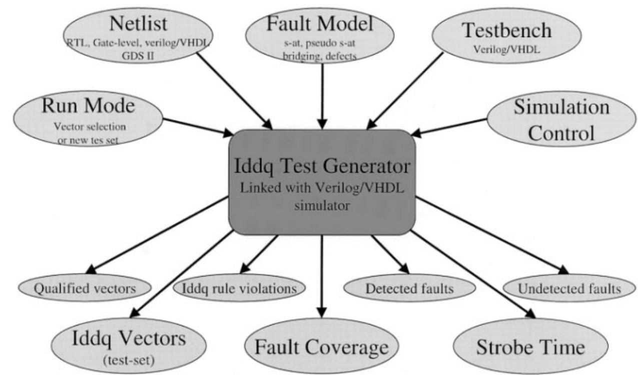

Fig. 14. Generalized overall flow of Iddq Test Generator.

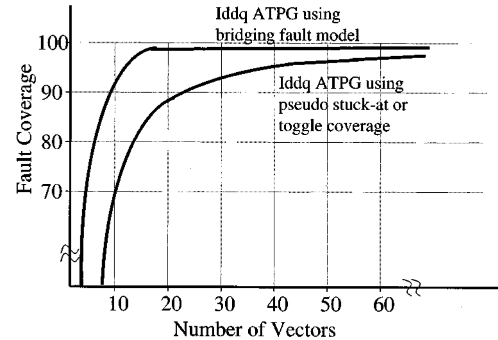

Fig. 15. Effectiveness of Iddq patterns generated using various fault models.

node in the transistor schematic is analyzed for each vector. This creates a large number of faults even for primitive gates. For example, a two-input NAND gate contains 25 faults in the CrossCheck model [73]. This extremely large fault set makes some Iddq test generators slow compared to the tools that use pseudo stuck-at fault model. To speed up the process, faults across cell boundary were considered. This reduces the number of faults, i.e., a 2-input NAND gate contains only 9 faults at the cell boundary and hence, simulation is faster. Another tool, PowerFault from System Science, can also work at the RTL netlist level. Although, it is not advisable to develop an Iddq test set using RTL netlist, running the tool at that level identifies Iddq rule violations (such as bus conflicts) in the early stage. Hence, this type of tool also becomes useful in the design process.

Although, in the present state of industry, deterministic Iddq patterns are used exclusively, research papers have also been published on using random or pseudo random patterns. The applicability of conventional built-in self-test patterns (based upon LFSR or MISR) using high-speed current measurement devices has been reported. Similarly, the use of a stuck-at test set (full test set or randomly selecting few vectors) has also been reported [73]. Detailed mathematical models have been published for fault coverage, length of random test sequence and fault escape probability [73].

Based upon the above discussion and various studies, it can be summarized that: *designing for Iddq testing requires careful considerations and it may impose some restrictions; necessary EDA tools are available to obtain Iddq vectors;*

the appropriate number of Iddq vectors to use must be determined by trading off cost-benefits, often it is around 20 Iddq vectors either selected from functional test set for maximum bridging coverage or generated by Iddq ATPG for bridging coverage.

V. IDDQ TESTING IN DEEP-SUBMICRON TECHNOLOGY

In last couple of years with minimum feature size shrink to 0.25 μm and below, concern has been expressed if Iddq testing will continue to be useful [74], [100]. Before stating a yes or no answer, the issue needs to be understood.

The theoretical basis of Iddq testing is based upon estimation of defect-free current in the circuit and then setting a limit (popularly, called as Iddq threshold) above which a circuit is considered defective. While some research papers have been published on methodology and tools to estimate defect-free current, in industry it is based upon the measurement on large number of devices [117]. Generally, close to 1 μA is considered as defect-free and any number from as low as 10 μA to as high as 100 μA being considered as threshold. Due to the law of large numbers, the distribution of this measured current is expected to be Gaussian. Due to statistical variations, IC's up to mean+3σ are considered defect-free. A limit much higher than mean+3σ is assumed, above which IC's are considered defective. This concept is illustrated in Fig. 16 (for illustration, distribution of defective current is also assumed Gaussian).

When the density functions of defect-free and defective current are separate from each other, the clear distinction between the good and the defective IC can be made. However, with technology shrink (increased sub-threshold leakage) and increasing number of gates in an IC, the mean value of the distribution of defect-free current increases and approaches the Iddq threshold limit (set from earlier technology). Just changing the threshold limit to a higher number does not resolve the issue because with high leakage in the circuit, change in defect-free and defective current is minuscule and unidentifiable. Proposals have been made to partition the circuit and to perform Iddq testing on one partition at a time. However, due to the increased design complexity, area overhead and performance penalty, this idea has not been used.

The separation between the distribution of defective and defect-free current is:

$$\begin{aligned}&\mathrm{Iddq}_{\mathrm{defect}} - \mathrm{Iddq}_{\mathrm{defect\text{-}free}}\\&\quad = (\text{Average Iddq}_{\mathrm{defect}} - 3\sigma\mathrm{Iddq}_{\mathrm{defect}})\\&\qquad - (\text{Average Iddq}_{\mathrm{defect\text{-}free}} + 3\sigma\mathrm{Iddq}_{\mathrm{defect\text{-}free}}) \quad (1)\end{aligned}$$

This concern becomes very clear from I_{off} data of pMOS and nMOS from 0.35-, 0.25-, and 0.18-μm technologies. I_{off} represents the steady state leakage. Iddq in the IC can be assumed as summation of all leakage, i.e., (# of $\mathrm{nMOS} \cdot n - I_{\mathrm{off}} + \#$ of $\mathrm{pMOS} \cdot p - I_{\mathrm{off}}$). The characteristic data (range) with I_{off} values for various technologies are given in Table 1. Table 1 provides the range of parameters by various

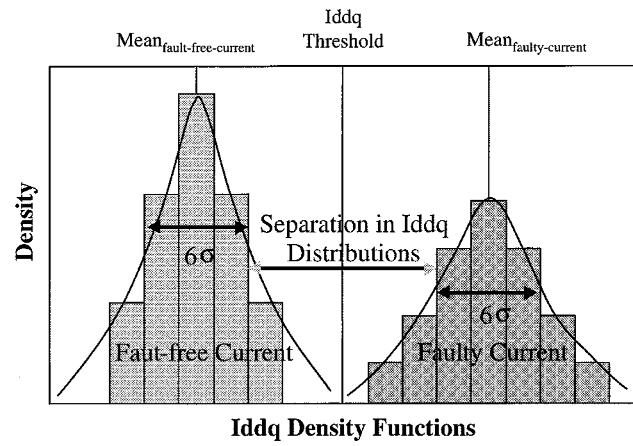

Fig. 16. Representation of fault-free and faulty Iddq density funtions.

Table 1
Characteristics of Various Technologies and Range of I_{off} Data [110]

Technology (µm)	Vdd (V)	T_{ox} (Å)	V_t (V)	I_{off} (pA/µm) NMOS	pMOS
0.8	5	150-100	0.8-0.7	0.01 – 0.05	0.005 – 0.02
0.6	5 – 3.3	100-80	0.75-0.65	0.05 – 0.5	0.01 - 0.2
0.5	5 – 2.5	90-70	0.7-0.6	0.1 - 2	0.1 – 1
0.35	3.3 – 2.5	80-60	0.65-0.55	0.5 - 10	0.1 - 10
0.25	3.3 – 1.8	70-50	0.6-0.5	6 - 60	0.5 - 24
0.18	2.5 – 1.8	55-35	0.55-0.45	40 - 600	20 -300

manufacturers for a specified minimum feature size. For example, for TSMC's (Taiwan Semiconductor Manufacturing Corporation's) 0.25 μm technology, T_{ox} is 55 Å, V_t is 0.55 V; nMOS I_{off} is 15 pA/μm for 2.5 V nMOS and 10 pA/μm for 3.3 V nMOS; while pMOS I_{off} is 7 pA/μm for 2.5 V pMOS and 0.5 pA/μm for 3.3 V pMOS. Further, for dual gate 0.25 μm process, nMOS I_{off} is 41 pA/μm and pMOS I_{off} is 3.5 pA/μm [75]. Also, within a technology, smaller channel length transistors show much higher leakage compared to longer channel length transistors (for example, in dual gate process, compared to 41 pA/μm for 0.25 μm transistor, nMOS I_{off} for 0.35 μm transistor is only 0.65 pA/μm.

Table 1 shows that I_{off} has increased 4–5 orders of magnitude from 0.8- to 0.18-μm technology. The increase in I_{off} can be understood by the transistor's V_G versus I_D transfer curve as shown in Fig. 17 [76]. The I_{off} is measured at $V_G = 0$ V, as shown in Fig. 17, for this transistor, it is 20 pA/μm for the saturated region and 4 pA/μm in the linear region. The subthreshold slope S_t (V_G versus I_D in the weak inversion region) is about 80 mv/decade of I_D. S_t is a function of the gate oxide thickness and the surface doping adjusted implant. The change in S_t is minimized by T_{ox} scaling and improved doping profiles. A $S_t \geq 100$ mV/decade indicates a leaky device, while a lower value results in low I_{off} for a given

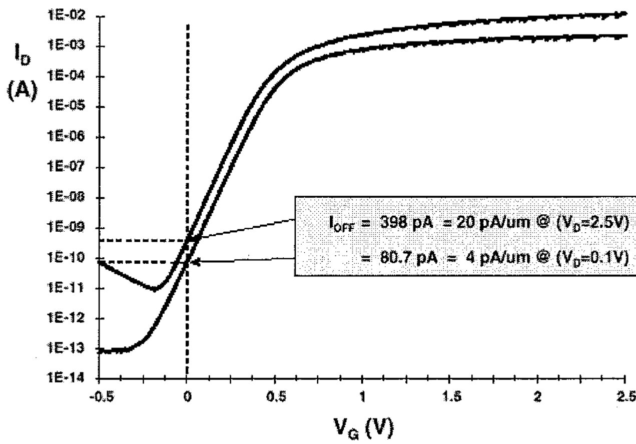

Fig. 17. Log I_D vs. V_G at saturated bias ($V_D = 2.5$ V) and linear bias ($V_D = 0.1$ V), for 20×4 μm nMOS transistor [76].

threshold voltage. S_t has been reported to change from about 75 in 0.35 μm technology to about 85 in 0.25 μm technology.

Another reason for increased $I_{\rm off}$ is drain-induced barrier lowering (DIBL) and gate-induced drain leakage (GIDL). DIBL moves the curve up and to the left as V_D increases while GIDL current shows-up as a hook in the transistor I_D versus V_G curve (Fig. 17). A quantitative model to explain these effects has been reported in [77]. In general, the nature of $I_{\rm off}$ with technology shrink is given as shown in Fig. 18.

Fig. 18 indicates that $I_{\rm off}$ has been less than 1 pA/μm for gate lengths of 0.35 μm or larger and it starts to increase exponentially at about 0.25 μm. The above discussion provides the underlying reason for increased $I_{\rm off}$ and concern for Iddq testing in deep submicron technologies. It is also clear from the above discussion that if I_D versus V_G curve in the linear and sub-threshold regions are moved down to the right, the problem of high leakage will go away, hence, Iddq testing will continue to be useful. Two mechanisms have been proposed [76], [101]: (a) reduced temperature and (b) substrate bias. A mathematical model to explain the effect of substrate bias on sub-threshold current is given in [86]. The quantitative data to illustrate the effect of temperature and substrate bias is shown in Fig. 19(a) and (b), respectively.

The dramatic reduction in $I_{\rm off}$ is clear from Fig. 19. As an example, 42 pA $I_{\rm off}$ at room temperature can be reduced to about 9 pA at 0°C, a reduction factor of about 4.5. Similarly, 9.6 nA $I_{\rm off}$ at $V_{\rm sub} = 0$ V can be reduced to about 2 pA at $V_{\rm sub} = -4$ V, a reduction factor of about 4400. It should be noted that beyond a certain point further decrease in $V_{\rm sub}$ increases $I_{\rm off}$, however, lowering both $V_{\rm sub}$ and V_D reduces $I_{\rm off}$. Using these methods, reduction-factor as high as 60 000 has been reported.

It is also worth mentioning that 1999 International Technology Roadmap for Semiconductors (ITRS) has predicted that the maximum $I_{\rm off}$ will be about 1 nA/μm for the technologies upto 0.18 μm; it will increase to about 3 nA/μm for 0.15–0.10-μm technologies and further increase to about 10 nA/μm for 0.07–0.05-μm technologies [119]. Indeed, the production process upto 0.18-μm technology and experimental device up to 0.10-μm technologies across various fabrication facilities show $I_{\rm off}$ values similar to

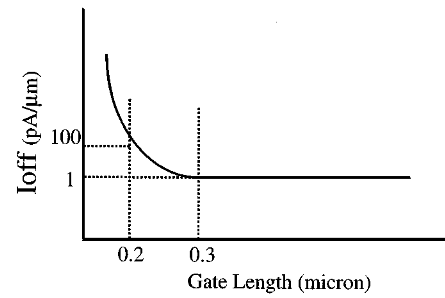

Fig. 18. Nature of $I_{\rm off}$ with geometry shrink.

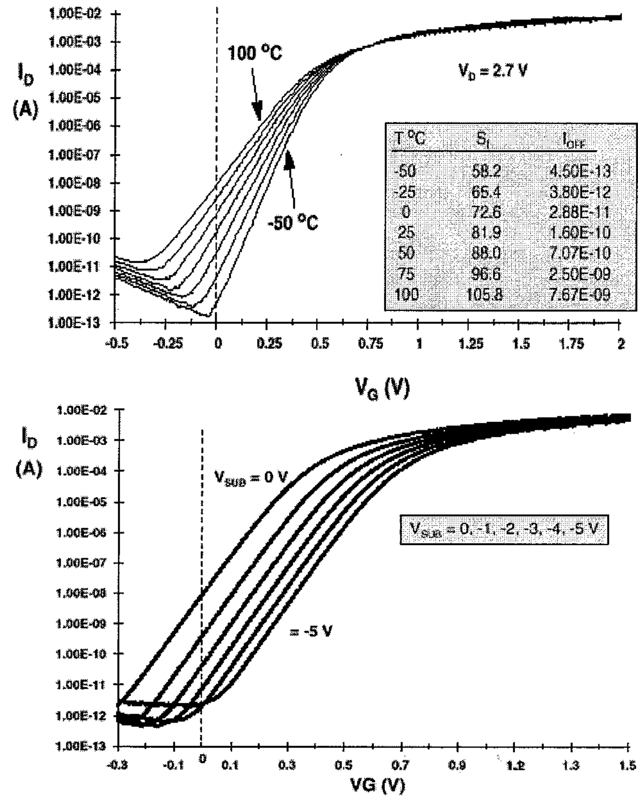

Fig. 19. The effect of (a) temperature and (b) substrate bias on linear and sub-threshold regions [76].

this prediction. Reference [119] also suggests a number of solutions for continuous use of Iddq testing. These solutions include substrate bias, lower temperature, and power-supply partitioning at chip level and the use of multiple power sources.

From the above discussion, it can be summarized that *the concern over leakage current* $(I_{\rm off})$ *has been over-stated, further with the methods such as substrate bias, lower Vdd and lower temperature, Iddq testing can be used successfully even in deep submicron technologies*.

VI. IDDQ TESTING FOR RELIABILITY SCREENING

Stress testing based upon voltage, temperature, humidity, vibration and power cycling is commonly used to accelerate early failures. A majority of semiconductor companies have a stress testing standards that are similar in nature with some variation in terms of temperature setting, relative humidity or time duration. In general, these accelerated qualification

Table 2
Accelerated Tests Used by 5-Semiconductor Manufacturers [78]

Test	TI	Micron	AMD	Matsushita	National
Static high Temperature life	150 C 168 hrs	----	125 C 2000 hrs	1000 hrs	----
Thermal Cycling	-65 to 150 C 500 cycles	-65 to 150 C 1000 cycles	-65 to 150 C 1000 cycles	-55 to 150 C 10 cycles	-65 to 150 C 1000 cycles
Autoclave	121 C 15 psig 240 hrs	121 C 15 psig 96 hrs	121 C 15 psig 168-500 hrs	121 C 30 psig 8 hrs	121 C 15 psig 500 hrs
Temperature Humidity Burn-in	85 C 85% RH 1000 hrs	85 C 85% RH 1000 hrs	85 C 85% RH 2000 hrs	85 C 85% RH 1000 hrs	85 C 85% RH 1000 hrs
Operating life	125 C 1000 hrs	150 C 1000 hrs	125 C 168 hrs	----	125 C 1000 hrs
High temperature Storage life	----	150 C 1008 hrs	125 C 2000 hrs	1000 hrs	150 C 1000 hrs
Low temperature life	----	-10 C 1008 hrs	-10 C 1000 hrs	-40 C 1000 hrs	----

tests contain some sort of burn-in test at 125°C to 150°C. Representative methods from 5 commercial companies are given in Table 2 [78].

The time duration in these qualification methods varies from 168 h (1/week) to 2000 h. Due to variations in test conditions, the model for reliability prediction also varies. Examples of reliability prediction models are given in [79]–[83]. Due to time consuming and expensive nature of these tests, alternatives have been sought which can screen infant mortality related failures quickly and at lower cost. Another motivation is that the methods such as Burn-in are destructive in nature [118], hence, it is very difficult to identify the original cause of failure by performing post burn-in failure analysis.

Most infant mortality related failures are due to defects such as gate-oxide shorts, pinholes, partial open, resistive bridging, etc., which may not cause a functional error at the time of testing, but degrade product lifetime. Thus, in a majority of cases, these defects are a reliability concern [118]. Since Iddq testing detects physical defects regardless of whether it causes a functional error, it has been thought-of as a potential method for reliability screening. Other motivations are small test time compared to burn-in, hence, shortened time–market; significantly less cost compared to burn-in; nondestructive nature of the test, hence, ease in failure analysis. It is worth emphasizing that fast detection and removal of infant mortality related failures is extremely desirable for early qualification of the process, which has tremendous impact on cost, profit and revenue of the company.

The exact motivation is explained through popular bathtub curve for reliability. Fig. 20 provides two views of the bathtub curve to illustrate the impact of Iddq testing based screening of early failures. The reason why burn-in is used is illustrated in Fig. 20(a). The product is qualified for market as soon as the FIT (failure-in-time, 1 FIT is defined as 1 failure in 10^9 device hours) rate is stable, burn-in provides considerable timesavings in flattening the curve and stabilizing the FIT rate. Figure 20(b), illustrates that similar time savings can be obtained over burn-in by Iddq testing based screening of the infant mortality failures. For example, if Iddq testing is used to supplement a reduced burn-in flow (for example only 24 hours instead of 1000 hours in qualification testing or 24 hours instead of 168 hours in production reliability screening), whatever time is reduced it shortens time–market. Thus, whatever comfort level is acceptable, it is still desirable to use some reduction in burn-in.

In the last couple of years, a number of IC manufacturers have conducted experimental studies in this direction. These studies also provide comparative data with burn-in based conventional reliability screening [13]–[15], [83], [84]. As an example, the results of one study are shown in Fig. 21 [14].

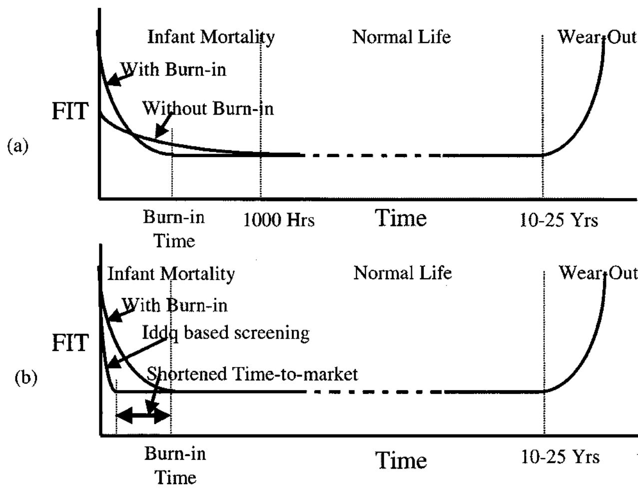

Fig. 20. Effect of Iddq testing on reliability curve: (a) nature of bathtub curve with no screening and with burn-in; (b) nature of bathtub curve with burn-in and with Iddq testing based screening.

In this study, a 0.5-μm 3.3-V technology, 140-K gates gate array based design in 240 pin plastic quad flat package (PQFP) was used. First all dies were tested for gross failures, the passed dies were divided into two groups. One group was exposed to conventional burn-in based test flow (4-V 150 °C high temperature operating life test), while the second group was exposed to logic test and stressed Iddq-test. The IC's that passed the stressed-Iddq test were subjected to burn-in and failures were observed at 24, 48, and 168 hours of burn-in.

Two important observations from this study were: a) stressed-Iddq testing at 40%–50% higher voltage (5.0 V for 3.3 V technology) was able to screen all gate-oxide infant mortality failures equivalent to 168 hours of burn-in; b) stressed-Iddq testing was able to screen about half of the via-defects, an additional 6 h of burn-in screening was required to detect all failures.

Similar conclusions were drawn regarding stressed-Iddq testing through studies on automotive ASIC, zero-hour burn-in experiment on i960 microprocessor [15] and SEMATECH sponsored project on 1 M-bits SRAM's [85]. Reference [85] also provides a risk assessment by analyzing the IC's which pass functional test but show high Iddq at nominal Vdd and at 40%–60% stress voltage, shown in Fig. 22.

Figure 22, shows that Iddq testing at nominal voltage (5 V) detected about 50% of post-burn-in failure, about 56% at 40% stress (7 V) and about 84% failures at 60% stress (8 V). It also shows that about 9.2% faulty IC (about 92 000 PPM) would have escaped if 60% stressed-Iddq testing had not been used.

Based upon above discussion and various studies, it is adequate to say that *stressed-Iddq testing can effectively screen infant mortality failures and can be used to reduce the burn-in time*.

VII. Iddq Measurement Methods

Since Iddq testing is based upon measurement of quiescent current, generally, it is performed at a slow speed. Many IC manufacturers use a 1–10 ms wait-time after the application of each Iddq vector before the actual measurement is done. The necessary requirement for Iddq testing is that all current spikes in the circuit due to switching activity have died-out, the 1–10 ms is sufficient time for this purpose. In the early years, current measurement equipment related issues were also part of the reasons for this long wait-time.

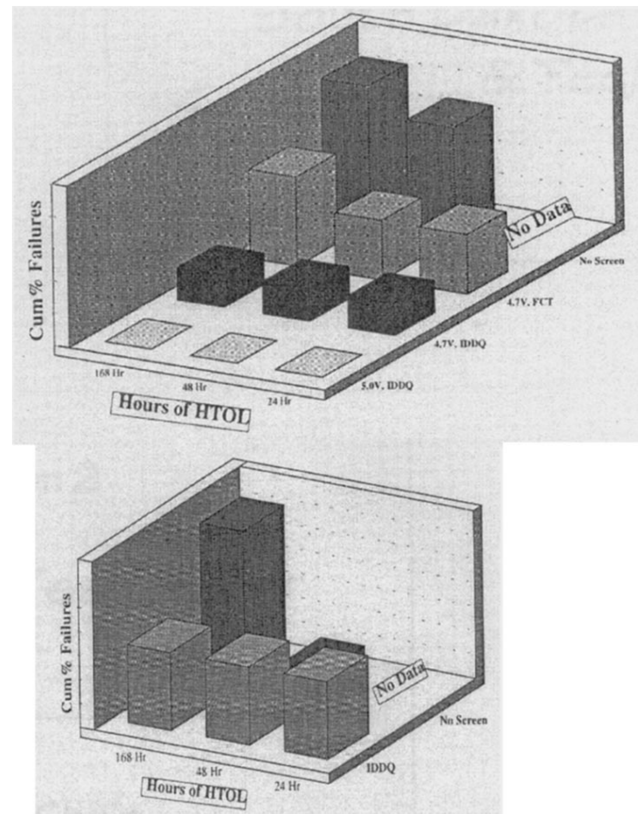

Fig. 21. Effectiveness of Iddq testing based screening: (a) Effectiveness for gate-oxide defects, after stressed-Iddq at 5.0 V no additional failure was observed in 168-hours of burn-in [14]; Effectiveness for via defects, after stressed-Iddq testing and 6-hours of burn-in no additional failures were detected by further burn-in [14].

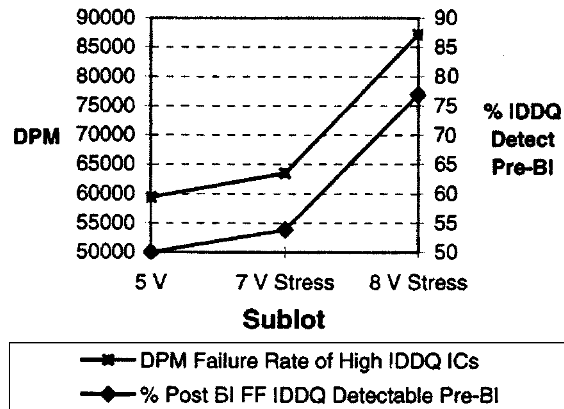

Fig. 22. Detection of post-burn-in functional failures by stressed-Iddq testing [85].

Because of the long wait-time, Iddq test time has been a major discussion topic. A large number of papers have been written to do this measurement at a faster speed. These papers

address both on-chip measurement as well as off-chip measurement; we will discuss these methods in separate subsections.

A. On-Chip Current Measurement

The main motivation of on-chip current measurement was to avoid delay due to measurement equipment. Additional reasons were to avoid LRC-drop across the current probe and hence improve the accuracy in measurement, to avoid mechanical limitations of commercially available current probes, and if high-speed measurement is possible then measure Iddq on every eligible vector. The general concern has been that a typical IC I/O pin may have about 100 Ω output impedance, driving a 50 pF, 100 Ω line. Simultaneous switching of multiple lines may draw upto 5 A current transient of about 10 ns pulse-width with an edge speed of 10 A/ns. A simple current probe offers significant loading at the power supply causes a large voltage drop across it and lacks in DC accuracy.

To overcome some of these issues, suggestions were made to use either an active monitor circuit in-between the device under test (DUT) and device power supply (DPS) or to use a special DPS near the test head with ultra low impedance cabling between the DUT and DPS [87]. These kinds of DPS were built in Sentry test systems to facilitate Iddq testing.

However, as mentioned above, issues remained unresolved and off-chip Iddq measurement remained slow. A significant research was conducted toward high-speed on-chip current monitors in the late 80's and early 90's. For on-chip current sensor (also, called as built-in current sensor or BICS), the first step was to partition the circuit into several sections, with each section containing a current sensor. The current sensor itself was made-up of a nonlinear resistance circuit breaker and a differential amplifier. The schematic of an early on-chip current sensor built at Carnegie Mellon University is shown in Fig. 23 [88].

In Fig. 23, in a fault-free situation, transistor T1 is on and T2 is off. In the presence of a defect, the circuit under test (CUT) draws high current. As the voltage at virtual ground increases, transistor T2 switches on and T1 switches off. This effectively isolates the CUT from power supply and works as circuit breaker. It should be noted that a third transistor T3 ensures that the circuit breaker will operate in the correct conducting state. This transistor restores the voltage at node 3; hence, it is designed to offer a high on resistance and allows only a small leakage current under the fault-free situation.

The differential amplifier compares the virtual ground with a reference voltage. This reference voltage is predetermined based upon the virtual ground voltage induced by the normal static current through the CUT. This differential amplifier is designed carefully to achieve the required switching resolution and to minimize the amplifier's offset sensitivity. The output of this amplifier is a pass/fail flag, identifying a fault-free/faulty circuit.

The circuit breaker in sensor shown in Fig. 23 can be removed to save both the hardware and performance penalty. One such example is shown in Fig. 24, which uses a diode in-

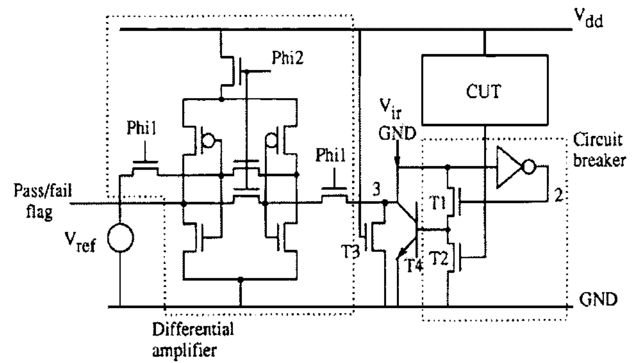

Fig. 23. Circuit schematics of an on-chip current sensor [88].

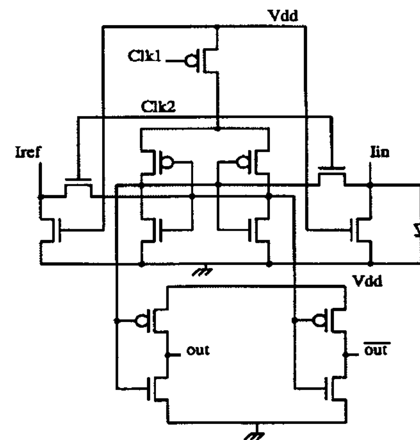

Fig. 24. On-chip current sensor without circuit breaker for fast response [89].

stead of a circuit breaker [89]. This diode is required to sink the large currents without significant change in voltage. The sensor shown in Fig. 24 develops a voltage across the parallel transistor-diode combination when abnormal input current is present. A sense amplifier converts this voltage into a logic output. In this circuit, an approximate drop of about 0.6 V to 0.65 V across the diode causes about a 10% to 15% decrease in the circuit speed, which is better than the sensor with circuit breaker.

Many innovative designs for on-chip current sensors have been proposed which provide fast response and very small voltage drop. Also, to enhance the measurement resolution of on-chip current sensors in large designs, proposals for circuit partitioning and the use of one sensor per partition have been made [108], [109]. However, on-chip sensors have not been used in actual products. The main reasons are:

1) Circuit partitioning requirement significantly increases design complexity.
2) Multiple on-chip sensors. For large IC's, one on-chip sensor is inadequate and multiple sensors result into significant hardware overhead.
3) A permanent loading on circuit power supply. Due to increased parasitic and loading, on-chip sensor result into significant performance penalty even during normal operation of the circuit [115].

In [90], simulation results have been reported to estimate the performance penalty, shown in Fig. 25. Although, the

simulation data shown in Fig. 25 is based upon inverter chains of various lengths, and the number also depends on the technology parameters, it illustrates the magnitude of degradation in circuit performance. Due to these reasons, *today, on-chip current sensors have only academic interest.*

B. Off-Chip Current Measurement

The most common method for current measurement is off-chip current measurement. Many semiconductor vendors perform Iddq testing through tester PMU (precision measurement unit). In this method, the device power-supply (DPS) supplies the transient current during sensitization; then an electromechanical relay on the load board switches from DPS to PMU to perform the measurement. After the measurement, DPS is switched back by the relay. Each switching of relay requires a delay equivalent to settling time and hence, in general, this testing is considerably slow.

Various AC and DC current probes are commercially available. In [91], a comprehensive discussion is given on the use of external current probes. Conceptually, a current probe can be used in between the DUT and power supply as shown in Fig. 26. The basic problem with such probing is the insertion inductance that is typically 10 nH to 50 nH. If a current pulse of an edge speed of about 10 A/ns were fed to a 10 nH inductive probe, it would cause about 100 V voltage drop across it. This simply means that such probing cannot be used.

One solution is to use an op-amp with sufficient gain, while keeping the current-sense resistor in its feedback loop, as shown in Fig. 27(a). This op-amp should be designed to compensate for the voltage drop across the sense resistor as well as be able to supply high transient currents to the circuit. Obviously, designing such a current sensor will be difficult and costly. The solution to this problem is to provide a shunt path for the transient current across the sense resistor. If this shunt path is provided by a diode, as shown in Fig. 27(b), it still would cause about a 0.6 V drop across it and hence cannot be used in production testing due to specified testing voltage.

To avoid this voltage drop across the shunt path, a FET bypass circuit can be used as shown in Fig. 27(c). This bypass transistor is ON only during the transient. Thus, when the transient is settled down, the current is passed through the sense resistor. To filter the high impedance noise at high frequencies, a small capacitor is added in between the sense circuit and the DUT, as shown in Fig. 27(d). It has been reported that about 2000 pF to 2500 pF capacitor and a 400 Ω to 500 Ω resistor provide an adequate bypass circuit. The only disadvantage with this method is that it causes a RC loading at the output, hence, the circuit takes more time to stabilize.

If the circuit in Fig. 27(d) is examined critically, one may realize that the resistance in the sense circuit is redundant. The only requirement in current sensing is a bypass circuit for the transient. By eliminating resistance, the testing speed can be improved significantly. This modification is shown in Fig. 28. In this circuit, as before, the FET is ON during the transient when DUT is drawing large current. Once transients are settled, the FET is OFF and capacitor C1 supplies the static current to DUT. The Idd is measured by the voltage drop across the FET. In this circuit, the value of capacitor C1

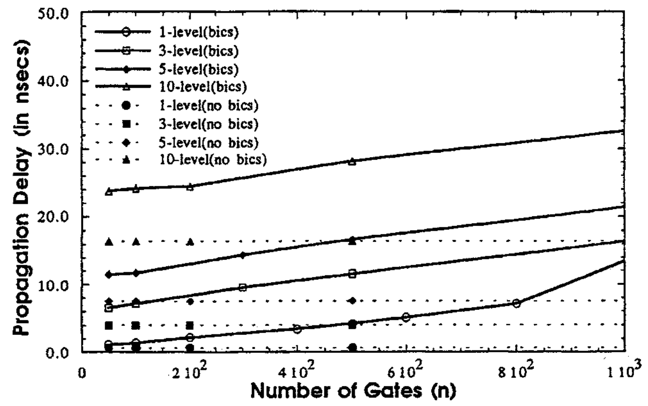

Fig. 25. Penalty on circuit performance due to on-chip current sensor [90]. Levels represent number of inverters in a path from primary input to primary output, for example, 1-level means one inverter, 3-level means three inverters, 5-level means five inverters, etc.

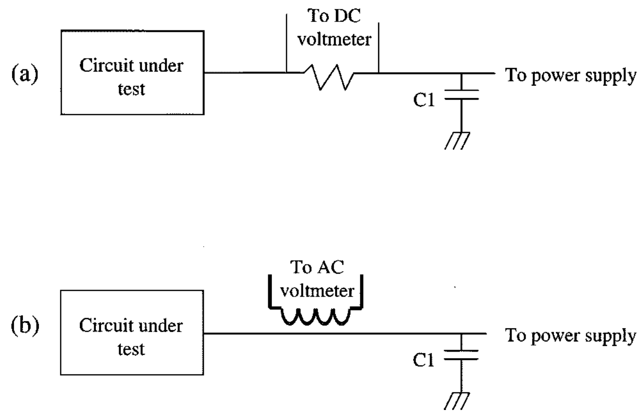

Fig. 26. Schematics of current measurement methods: (a) AC current probe [91]; (b) DC current probe [91].

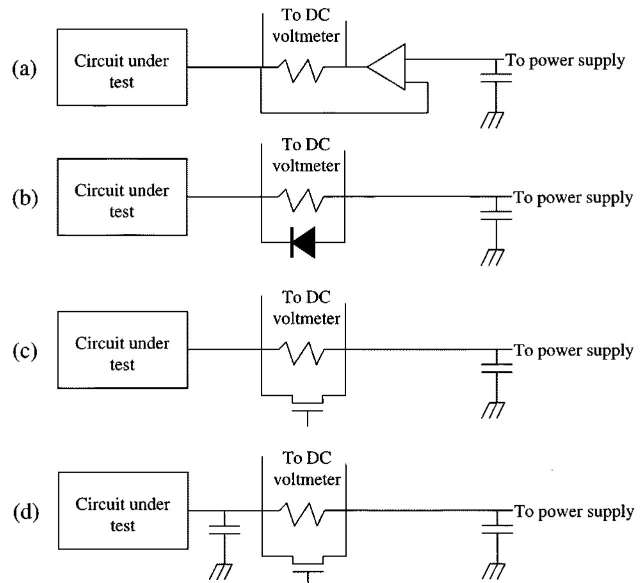

Fig. 27. Current measurement techniques using external probe [91].

is critical, it should be chosen such that in the fault-free circuit, it will keep Vdd to the specified testing voltage at-least until the measurement is done.

The circuit shown in Fig. 28 can perform Iddq testing up to 10–50 KHz. Reference [92] analyzes the design limitations of this method at high frequencies. To increase the measurement speed, the size of capacitor in the bypass circuit was identified as one of the key item. Reference [92] also provides a significantly improved method for current measurement at high speed (up to 1 MHz), the circuit schematics is shown in Fig. 29 [92].

The method shown in Fig. 29 uses nonfeedback voltage source for setting the fixed current and threshold voltage by clamping voltage sources VL and VH, while a bridge-type diode circuit allows fast switching between small and large current values. The power-supply current to DUT is supplied through diode D2 or D4. The high-speed current source provides current to the bypass capacitor CL to recover the voltage drop during peak current. It was reported that undershoot of only 82 mV was achieved by this circuit during a peak current of 500 mA. This circuit also provided a relatively constant settling time with varying loads, peak current values from 100 mA to 500 mA and peak width from 50 ns to 200 ns. Using this method, the size of bypass capacitor was reduced to 1000 pF and Iddq testing upto 1 MHz was achieved.

It is interesting to observe that in addition to the Precision Measurement Unit (PMU), a number of ATE manufacturers have provided current measurement facility since the mid 1980s. These additional facilities can perform Iddq testing upto 100 KHz frequency. Since most IC manufacturer use only a few Iddq measurements, they use the tester PMU instead of dedicated Iddq test unit in production testing [117]. Iddq testing with a large number of vectors and high-speed measurement method has been rarely done in production environment.

Based upon above discussion, it is adequate to say that *off-chip Iddq measurement methods are adequate; the on-chip current sensors impose design constraints, which limit their use*.

VIII. Comments on Iddq Test Economics

The general acceptance of any test method depends upon its economic benefits. For the most part, the cost of Iddq testing is added test generation time, added test execution time, perhaps higher diagnostics cost, very little to no area overhead and the cost of rejected parts which might otherwise be sold. The benefits include reduced IC cost with less DFT overhead, early detection of failure, improved product quality leading to less return and less warranty costs.

With respect to test generation, there is an on-going debate on how many Iddq vectors should be used. However, as discussed in Section IV and shown in Fig. 15, about 20-25 vectors from Iddq ATPG or selected from the functional test-set under bridging fault model provide the best benefit in least

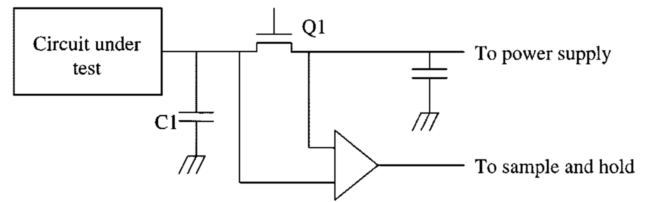

Fig. 28. External current-sense circuit to avoid RC loading at the output [91].

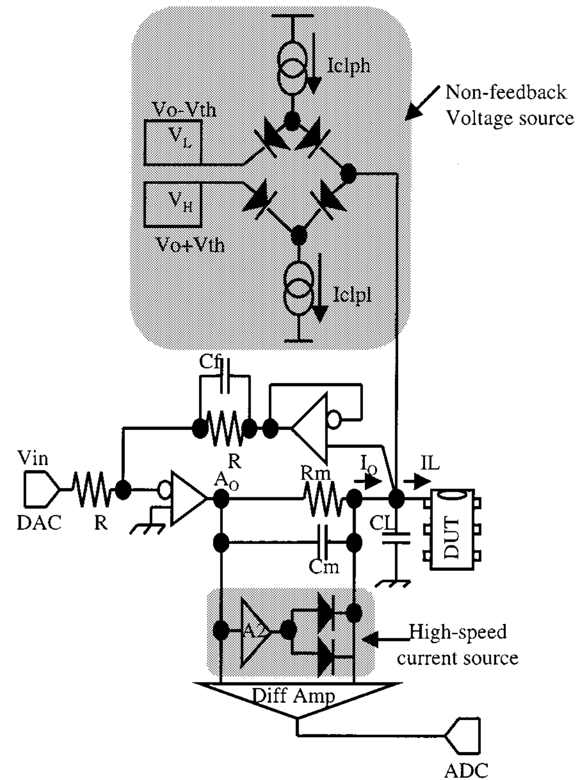

Fig. 29. External current-sense circuit for fast response [92].

cost. Chapter 3 in [73] provides additional analysis of various types of vectors and their efficiency including stuck-at vectors, random, pseudorandom and Iddq ATPG vectors.

Computing the economics of Iddq testing for an IC customer involves computing the benefits from less IC dropout during board manufacturing as offset against the increased cost of rigorous testing for in-coming parts. An IC customer can argue that improved quality should cost no more. In in-coming quality inspection, if the IC failure rate is very low, Iddq testing may not be warranted as it will provide very little advantage. On the other hand, a high dropout rate will provide a reason to request that the IC vendor put an Iddq test program in place.

From the IC vendor perspective, many of the issues are the same. The benefit in improved product quality must outweigh the increased cost of Iddq testing. One significant factor is the decreased yield. Since, Iddq testing will also screen devices that pass logic test; the Iddq test dropout represents a loss of IC yield to the IC vendor. This is to be expected as Iddq testing detect defects that may not

be detectable by other tests. Qualitatively, the cost-quality relationship of Iddq testing is similar to as shown in Fig. 30 [99].

As discussed in Section III and data shown in Fig. 11, there are devices that fail Iddq test but pass all logic tests. For the IC vendor, the question is—are these devices defective? Evidence indicates that they are. However, are these bad IC? The answer depends on the customer perception of bad. Users of IC's in defense and aerospace industry would consider any IC with a defect as being bad because of very stringent view on quality and reliability [116]. On the other hand, users of IC's in the low-cost toy industry would be likely to have a relaxed view of IC quality; here, cost would be more important.

Yield can be defined in two ways: a) True Yield, percentage of devices produced which have no defect, i.e., $Y_T =$ Non-defective IC's/Total IC's and b) Customer-visible Yield $Y_C =$ Customer-good IC's/Total IC's. The difference between true yield and customer yield is one of the costs of Iddq testing. Computing the cost and benefits of Iddq testing involves the following:

1) Computing the cost of implementing Iddq testing in increased test time and complexity and additional test development effort.
2) Computing the savings from Iddq testing in achieving higher fault coverage with less test generation and fault simulation effort and in decreased reliance on expensive structural DFT techniques.
3) Computing the benefits of Iddq testing in the shipment of fewer customer-bad IC's, thus, reduced warranty costs and increased customer satisfaction, psychological factors and company image, etc.
4) Computing the costs of Iddq testing in potentially scrapping customer-good IC's, thus, the cost of decreased yield.

The first two items can be determined by examining a particular company's IC design and manufacturing process. However, items #3 and #4 are not straightforward as these two items contain an embedded trade-off. For a defense or aerospace customer shipping a potentially bad device is unacceptable, while for low-end consumer goods, scrapping a potentially good device is net loss in profit margin. Hence, it is advised that *before putting Iddq test program in-place, its cost-benefit factors should be carefully examined for each individual situation.*

IX. Continuing Research Topics

Although, a significant number of companies have adopted Iddq testing as part of their standard test-flow, it is still a relatively new method compare to other test methods in the voltage environment. Thus, Iddq testing is a major target for on-going research.

One of the on-going research topics is to identify its relationship with delay fault testing. It has been identified that in-addition to defects discussed in Section II, Iddq testing is also useful for detecting defects such as resistive vias, resistive bridges, partial open, capacitive coupling, etc. Many

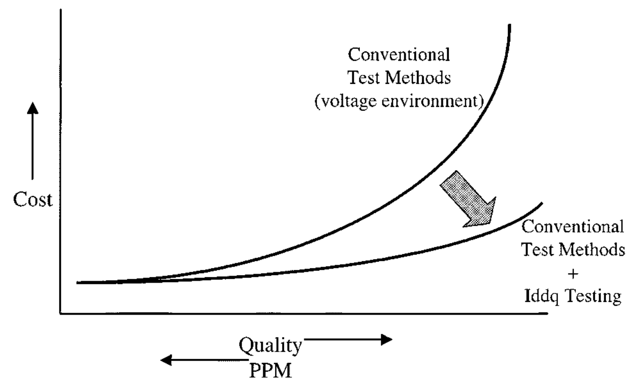

Fig. 30. Qualitative relationship between cost and quality of Iddq testing [99].

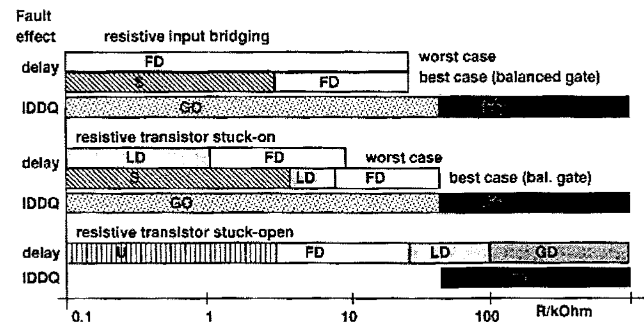

Fig. 31. Resistive faults and their impact on delay and Iddq for an AND circuit [93]. The notations are: $S =$ stuck-at; FD = fine delay (<100% of gate delay); LD = large delay (>100% of gate delay); GO = gross overcurrent (>100 μA); FO (shown in dark) = fine overcurrent (<100 μA); $u =$ untestable.

of these defects result into delay faults. In [93], an attempt was made to relate delay and Iddq effect of resistive faults as shown in Fig. 31. The question of its effectiveness for delay faults is also under investigation as part of the recent SEMATECH experiment and a few other studies.

A somewhat related topic is very-low-voltage (VLV) testing and its relationship to Iddq testing. VLV testing is based upon the fact that for a given technology, a circuit's switching speed is directly proportional to Vdd and thus, a delay fault can easily manifest itself as a functional failure at low voltage. This topic is a subject of on-going research at Stanford University [94], [103].

Another open item is the effectiveness of Iddq testing for asynchronous circuits. In a recent report, a simple design-for-test scheme has been described to address the self-timing issues in asynchronous circuits [112].

Another research topic is the use of Iddq testing in defect diagnosis. This topic is under investigation in various modes by different researchers. Logic fault dictionaries coupled with Iddq information [95], expert systems based upon Iddq values [96], [97] have been proposed. Other researchers have proposed using Iddq testing with commonly used failure analysis techniques such as optical emission, photon emission microscopy, infrared imaging and liquid crystal analysis [113], [114].

Yet another research topic is transient current measurement based analysis and fault detection. Using the

measurement of transient current (this topic is referred as Iddt testing), as a reference to determine a fault has also been proposed [104].

Research to identify defective IC's based upon current signature has been proposed [98]. The elaborate use of Iddq measurement data by rearranging the measured values and forming a reference signature has been actively researched recently at Carnegie Mellon University [98], [105], [106] and Sandia Labs [40], [107]. At the present time, this topic has been targeted for IC characterization testing as well as failure analysis and fault diagnosis.

X. Concluding Remarks

In this paper we tried to capture the basic knowledge on Iddq testing. A brief discussion and sample data are given on various topics related to Iddq testing. In a one sentence summary, Iddq testing can be described as a *low-cost, high-quality, supplemental* test. Few major conclusions are as follows:

1) Iddq testing is not a panacea. It is a high quality *supplemental test* that can improve overall fault coverage without incurring significant test development cost.
2) Iddq testing is very effective in detection of bridging and gate-oxide defects; however, it may not detect an open fault.
3) Some design rules are necessary to make a design suitable for Iddq testing. The majority of these rules are simply good design practices.
4) Simple design-for-test effort to switch-off static current dissipating logic can significantly enhance the design suitability for Iddq testing.
5) A variety of Iddq test generation and simulation tools are available. An Iddq tool running at RTL can also help in the design process. About 20 Iddq vectors with the highest coverage through fault grading under bridging model is a suggested cut-off point on the cost-benefit trade-off curve.
6) With each technology shrink, $I_{\rm off}$ increases. However, Iddq testing can still be used successfully in deep submicron technologies with a method such as substrate bias, lower Vdd and lower temperature. For very large system-on-a-chip devices, the simplest method to perform Iddq testing is to do it on one core at a time.
7) Iddq testing can detect infant mortality related defects. Stressed Iddq testing is particularly useful in screening infant mortality failures and can be used to reduce the burn-in time.
8) Many designs for on-chip current sensors are available. However, due to increased design complexity, area and performance penalty, at this time on-chip current sensors have little use in actual chips.
9) The methods for off-chip current sensing have been reasonably well understood. These methods are adequate to provide measurement speeds up to 1 MHz.
10) There are still a number of open research topics. Particular areas of interest are effectiveness of Iddq testing for delay faults and Iddq based diagnosis.

Acknowledgment

Without the continued support and encouragement from R. Sauer, H. Yamoto and S. Sugamori this paper would not have materialized. The constant encouragement from Jim Calder was invaluable. Finally, the comments and suggestions from reviewers were very useful in improving the quality of this paper.

References

[1] M. W. Levi, "CMOS is most testable," in *Int. Test Conf.*, 1981, pp. 217–220.
[2] Y. K. Malaiya and S. Y. H. Su, "A new fault model and testing technique for CMOS devices," in *Int. Test Conf., ITC, 1982*, 1982, pp. 25–34.
[3] J. M. Acken, "Testing for bridging faults (shorts) in CMOS circuits," in *Design Auto. Conf.*, 1983, pp. 717–718.
[4] R. Y. Li, S. C. Diehl, and S. Harrison, "Power supply noise testing of VLSI chips," in *Int. Test Conf.*, 1983, pp. 366–369.
[5] D. Baschiera and B. Courtois, "Testing CMOS: A challenge," in *VLSI Design*, Oct. 1984, pp. 58–62.
[6] M. E. Turner, D. G. Leet, R. J. Prilik, and D. J. McLean, "Testing CMOS VLSI: Tools, concepts and experimental results," in *Int. Test Conf.*, 1985, pp. 322–328.
[7] C. F. Hawkins and J. Soden, "Electrical characteristics and testing for gate oxide shorts in CMOS ICs," in *Int. Test Conf.*, 1985, pp. 544–555.
[8] Philips HCMOS Designer's Guide, p. 140, Jan. 1986.
[9] E. I. Muehldorf, "A quality measure for LSI components," *IEEE J. Solid State Circuits*, pp. 291–297, Oct. 1974.
[10] R. L. Wadsack, "Fault modeling and logic simulation of CMOS and MOS integrated circuits," *Bell Syst. Tech. J.*, pp. 1449–1488, May–June 1978.
[11] J. Shen, W. Maly, and F. Ferguson, "Systematic characterization of physical defects for fault analysis of MOS IC cells," in *Int. Test Conf.*, 1984, pp. 390–399.
[12] H. Walker and S. Director, "VLASIC: A catastrophic fault yield simulator for integrated circuits," *IEEE Trans. Computer-Aided Design*, pp. 114–130, Jan. 1986.
[13] S. R. Mallarapu and A. J. Hoffman, "Iddq testing on a custom automotive IC," *IEEE J. Solid State Circuits*, pp. 295–299, Mar. 1995.
[14] T. Barrette, V. Bhide, K. De, M. Stover, and E. Sugasawar, "Evaluation of early failure screening methods," in *Proc. IEEE Int. Workshop on Iddq Testing*, 1996, pp. 14–17.
[15] T. R. Henry and T. Soo, "Burn-in elimination of a high volume microprocessor using Iddq," in *Int. Test Conf.*, 1996, pp. 242–249.
[16] F. Zarrinfar and R. Rajsuman, "Automated Iddq testing from CAD to manufacturing," in *Proc. IEEE Int. Workshop on Iddq Testing*, 1995, pp. 48–51.
[17] G. R. Case, "Analysis of actual fault mechanisms in CMOS logic gates," in *IEEE Design Auto. Conf.*, 1976, pp. 265–270.
[18] J. Galiay, Y. Crouzet, and M. Vergniault, "Physical versus logical fault models in MOS LSI circuits, impact on their testability," in *Int. Symp. Fault Tolerant Computing*, 1979, pp. 195–202.
[19] F. Fantini and C. Morandi, "Failure modes and mechanisms for VLSI ICs—A review," in *IEE Proc. Part G*, June 1985, pp. 74–81.
[20] W. Maly, "Modeling of lithography related yield losses for CAD of VLSI circuits," *IEEE Trans. Computer-Aided Design*, pp. 166–177, July 1985.
[21] K. G. Kemp, K. F. Poole, and D. F. Frost, "The effects of defects on early failure of metal interconnects," *IEEE Trans. Reliability*, pp. 26–29, Apr. 1990.
[22] K. L. Kodandapani and D. K. Pradhan, "Undetectability of bridging faults and validity of stuck-at fault test sets," *IEEE Trans. Comput.*, pp. 55–59, Jan. 1980.
[23] M. Karpovsky, "Universal tests for detection of input/output stuck-at and bridging faults," *IEEE Trans. Comput.*, pp. 1194–1198, Dec. 1983.

[24] S. Xu and S. Y. H. Su, "Detecting I/O and internal feedback bridging faults," *IEEE Trans. Computers*, pp. 553–557, June 1985.

[25] S. D. Millman and J. M. Acken, "Special application of the voting model for bridging faults," *IEEE J. Solid State Circuits*, pp. 263–270, Mar. 1994.

[26] S. D. Millman and J. P. Garvey, "An accurate bridging fault test pattern generator," in *Int. Test Conf.*, 1991, pp. 411–418.

[27] Y. K. Malaiya, A. P. Jayasumana, and R. Rajsuman, "A detailed examination of bridging faults," in *Int. Conf. on Computer Design*, 1986, pp. 78–81.

[28] M. E. Zaghloul and D. Gobovic, "Fault modeling of physical failures in CMOS VLSI circuits," *IEEE Trans. Circuits and Systems*, pp. 1528–1543, Dec. 1990.

[29] R. R. Montanes, E. M. J. G. Bruls, and J. Figueras, "Bridging defect resistance measurement in a CMOS process," in *Int. Test Conf.*, 1992, pp. 892–899.

[30] M. Favalli, M. Dalpasso, P. Olivo, and B. Ricco, "Analysis of resistive bridging fault detection in BiCMOS digital ICs," in *IEEE Trans. VLSI*, Sept. 1993, pp. 342–355.

[31] C. F. Hawkins and J. M. Soden, "Reliability and electrical properties of gate oxide shorts in CMOS ICs," in *ITC*, 1986, pp. 443–451.

[32] S. Holland, I. C. Chen, T. P. Ma, and C. Hu, "On physical models for gate oxide breakdown," *IEEE Electron Device Lett.*, pp. 302–305, Aug. 1984.

[33] A. Bhattacharyya, J. D. Reimer, and K. N. Rits, "Breakdown voltage characteristics of thin oxides and their correlation to defects in the oxides as observed by the EBIC technique," *IEEE Electron Device Letts.*, pp. 58–60, Feb. 1986.

[34] M. Syrzycki, "Modeling of gate oxide shorts in MOS transistors," *IEEE Trans. CAD*, pp. 193–202, Mar. 1989.

[35] R. R. Montanes, J. A. Segura, V. H. Champac, J. Figueras, and J. A. Rubio, "Current vs. logic testing of gate oxide short, floating gate and bridging failures in CMOS," in *Int. Test Conf.*, 1991, pp. 510–519.

[36] J. Soden and C. F. Hawkins, "Test considerations for gate oxide shorts in CMOS ICs," *IEEE Design and Test*, pp. 56–64, Aug. 1986.

[37] M. Jacomet and W. Guggenbuhl, "Layout dependent fault analysis and test synthesis for CMOS circuits," *IEEE Trans. CAD*, pp. 888–899, June 1993.

[38] W. Maly, "Realistic fault modeling for VLSI testing," in *IEEE Design Auto. Conf.*, 1987, pp. 173–180.

[39] P. S. Moritz and L. M. Thorsen, "CMOS circuit testability," *IEEE J. Solid State Circuits*, pp. 306–309, Apr. 1986.

[40] C. L. Henderson and J. M. Soden, "Signature analysis for IC diagnosis and failure analysis," in *Int. Test Conf.*, 1997, pp. 310–318.

[41] N. K. Jha, "Multiple stuck-open fault detection in CMOS logic circuits," *IEEE Trans. Comput.*, pp. 426–432, Apr. 1988.

[42] S. M. Reddy and M. K. Reddy, "Testable realization for FET stuck-open faults in CMOS combinational circuits," *IEEE Trans. Comput.*, pp. 742–754, Aug. 1986.

[43] D. L. Liu and E. J. McCluskey, "CMOS scan path IC design for stuck-open fault testability," *IEEE J. Solid State Circuits*, pp. 880–885, Oct. 1987.

[44] A. P. Jayasumana, Y. K. Malaiya, and R. Rajsuman, "Design of CMOS circuits for stuck-open fault testability," *IEEE J. Solid State Circuits*, pp. 58–61, Jan. 1991.

[45] S. D. Sherlekar and P. S. Subramanian, "Conditionally robust two-pattern tests and CMOS design for testability," *IEEE Trans. Computer-Aided Design*, pp. 325–332, Mar. 1988.

[46] C. Landrault and S. Pravossoudovitch, "Hazard effect on stuck-open fault testability," in *European Test Conf.*, 1989, pp. 201–207.

[47] R. David, S. Rahal, and J. L. Rainard, "Some relationships between delay testing and stuck-open testing in CMOS circuits," in *European Design Auto Conf.*, 1990, pp. 339–343.

[48] W. Maly, P. K. Nag, and P. Nigh, "Testing oriented analysis of CMOS IC's with Opens," in *Int. Conf. on Computer Design*, 1988, pp. 344–347.

[49] J. M. Soden, R. K. Treece, M. R. Taylor, and C. F. Hawkins, "CMOS IC stuck-open fault electrical effects and design considerations," in *Int. Test Conf.*, 1989, pp. 423–430.

[50] M. Renovell and G. Cambon, "Electrical analysis and modeling of floating gate faults," *IEEE Trans. Computer-Aided Design*, pp. 1450–1458, Nov. 1992.

[51] V. H. Champac, A. Rubio, and J. Figueras, "Electrical model of the floating gate defect in CMOS ICs: Implications on Iddq testing," *IEEE Trans. Computer-Aided Design*, pp. 359–369, Mar. 1994.

[52] P. C. Maxwell, R. C. Aitken, V. Johansen, and I. Chiang, "The effectiveness of Iddq, functional and scan tests: How many fault coverages do we need?," in *Int. Test Conf.*, 1992, pp. 168–177.

[53] K. Sawada and S. Kayano, "An evaluation of Iddq versus conventional testing for CMOS sea-of-gate ICs," in *Int. Test Conf.*, 1992, pp. 158–167.

[54] P. C. Wiscombe, "A comparison of stuck-at fault coverage and Iddq testing on defect levels," in *Int. Test Conf.*, 1993, pp. 293–299.

[55] C. H. Chen and J. A. Abraham, "High quality tests for switch-level circuits using current and logic test generation algorithms," in *Int. Test Conf.*, 1991, pp. 615–622.

[56] T. Storey, W. Maly, J. Andrews, and M. Miske, "Stuck fault and current testing comparison using CMOS chip test," in *Int. Test Conf.*, 1991, pp. 311–318.

[57] P. Franco, W. D. Farwell, R. L. Stokes, and E. J. McCluskey, "An experimental chip to evaluate test techniques chip and experimental design," in *Int. Test Conf.*, 1995, pp. 653–662.

[58] S. C. Ma, P. Franco, and E. J. McCluskey, "An experimental chip to evaluate test techniques experiment results," in *Int. Test Conf.*, 1995, pp. 663–672.

[59] P. Nigh, W. Needham, K. Butler, P. Maxwell, R. Aitken, and W. Maly, "So what is an optimal test mix? A discussion of the SEMATECH method experiment," in *Int. Test Conf.*, 1997, pp. 1037–1038.

[60] P. Nigh, D. Vallett, J. Wright, F. Motika, D. Forlenza, R. Kurtulik, and W. Chong, "Failure analysis of timing and Iddq only failures for the SEMATECH test methods experiment," in *Int. Test Conf.*, 1998, pp. 43–52.

[61] A. Chan, D. Lam, W. Tan, and S. Y. Khim, "Electrical failure analysis in high density DRAMs," in *IEEE Int. Workshop on Memory Technology, Design and Testing*, 1994, pp. 26–31.

[62] J. Khare, S. Griep, H. D. Oberle, W. Maly, D. S. Landsiedel, U. Kollmer, and D. M. H. Walker, "Key attributes of an SRAM testing strategy required for effective process monitoring," in *IEEE Int. Workshop on Memory Testing*, 1993, pp. 84–89.

[63] C. F. Hawkins, J. M. Soden, R. R. Fritzemeier, and L. K. Horning, "Quiescent power supply current measurement for CMOS IC defect detection," *IEEE Trans. Indust. Electron.*, pp. 211–218, May 1989.

[64] K. J. Lee and M. A. Breuer, "Design and test rules for CMOS circuits to facilitate Iddq testing of bridging faults," *IEEE Trans. Computer-Aided Design*, pp. 659–670, May 1992.

[65] T. J. Chakraborty, S. Bhawmik, R. Bencivenga, and C. J. Lin, "Enhanced controllability for Iddq test sets using partial scan," in *IEEE Design Automation Conf.*, 1991, pp. 278–281.

[66] M. Colwell, R. Rajsuman, Z. Sarkari, and R. Abrishami, "Switchable Pull-Ups and Pull-Downs for Iddq Testing of Integrated Circuits," U.S. Patent No. 5 644 251, July 1, 1997 and Patent No. 5 670 890, Sept. 23, 1997.

[67] P. Nigh and W. Maly, "Test generation for current testing," *IEEE Design and Test*, pp. 26–38, Feb. 1990.

[68] S. Chakravarty and P. J. Thadikaran, "Simulation and generation of Iddq tests for bridging faults in combinational circuits," in *IEEE VLSI Test Symp.*, 1993, pp. 25–32.

[69] E. Isern and J. Figueras, "Test generation with high coverages for quiescent current test of bridging faults in combinational circuits," in *Int. Test Conf.*, 1993, pp. 73–82.

[70] K. J. Lee, C. A. Njinda, and M. A. Breuer, "SWiTEST: A switch level test generation system for CMOS combinational circuits," *IEEE Trans. Computer-Aided Design*, pp. 625–637, May 1994.

[71] J. P. Cusey and J. H. Patel, "BART: A bridging fault test generator for sequential circuits," in *Int. Test Conf.*, 1997, pp. 823–832.

[72] W. Mao, R. K. Gulati, D. Goel, and M. Ciletti, "Quietest: A quiescent current testing methodology for detecting leakage faults," presented at the Int. Conf Computer Aided Design, 1990, pp. 280–283.

[73] R. Rajsuman, *Iddq Testing for CMOS VLSI*: Artech House Inc., 1995, ISBN 0-89 006-726-0.

[74] T. W. Willams, R. H. Dennard, R. Kapur, and R. Mercer, "Iddq test: Sensitivity analysis of scaling," in *Int. Test Conf.*, 1996, pp. 786–792.

[75] *Proc. TSMC Technology Workshop*, 1997.

[76] A. Keshavarzi, K. Roy, and C. F. Hawkins, "Intrinsic leakage in low power deep submicron CMOS ICs," in *Int. Test Conf.*, 1997, pp. 146–155.

[77] A. Ferre and J. Figueras, "Iddq characterization in submicron CMOS," in *Int. Test Conf.*, 1997, pp. 136–145.

[78] L. T. Nguyen, R. H. Y. Lo, A. S. Chen, and J. G. Belani, "Molding compound trends in a denser packaging world: Qualification tests and reliability concerns," *IEEE Trans. Reliability*, vol. 42, no. 4, pp. 518–535, Dec. 1993.

[79] *Handbook of Reliability Data for Components Used in Telecommunication Systems*, British Telecom, Jan. 1987. issue 4.

[80] *Standard Reliability Table for Semiconductor Devices*, Mar. 1985.

[81] *Collection of Reliability Data from CNET*, 1983.

[82] *Reliability and Quality Specification Failure Rates of Components*, SN29500, 1986.

[83] S. McEuen, "Reliability benefits of Iddq," *J. Electronic Testing*, pp. 327–335, Dec. 1992.

[84] R. Kawahara, O. Nakayama, and T. Kurasawa, "The effectiveness of Iddq and high voltage stress for burn-in elimination," in *IEEE Int. Workshop on Iddq Testing*, 1996, pp. 9–13.

[85] A. W. Righter, C. F. Hawkins, J. M. Soden, and P. Maxwell, "CMOS Iddq reliability indicators and latent defect detection," in *Int. Test Conf.*, 1998, pp. 194–203.

[86] M. J. Chen and J. S. Ho, "A three parameters only MOSFET subthreshold current CAD model considering back-gate bias and process variation," *IEEE Trans. Computer-Aided Design*, pp. 343–352, Apr. 1997.

[87] C. Crapuchettes, "Testing CMOS Idd on large devices," in *ITC*, 1987, pp. 310–315.

[88] W. Maly and P. Nigh, "Built-in current testing—Feasibility study," in *Int. Conf. on Computer Aided Design*, 1988, pp. 340–343.

[89] T. L. Shen, J. C. Daly, and J. C. Lo, "A 2-ns detecting time, 2 μm CMOS built-in current sensing circuit," *IEEE J. Solid State Circuits*, pp. 72–77, Jan. 1993.

[90] S. M. Menon, Y. K. Malaiya, A. P. Jayasumana, and Q. Tong, "The effect of built-in current sensors (BICS) on operational and test performance," in *Int. Conf. on VLSI Design*, 1994, pp. 187–190.

[91] M. Keating and D. Meyer, "A new approach to dynamic Idd testing," in *Int. Test Conf.*, 1987, pp. 316–321.

[92] K. Isawa and Y. Hashimoto, "High-speed Iddq measurement circuit," in *Int. Test Conf.*, 1996, pp. 112–117.

[93] H. T. Vierhaus, W. Meyer, and U. Glaser, "CMOS bridges and resistive transistor faults: Iddq versus delay effects," in *Int. Test Conf.*, 1993, pp. 83–91.

[94] H. Hao and E. J. McCluskey, "Very-low voltage testing for weak CMOS logic ICs," in *Int. Test Conf.*, 1993, pp. 275–284.

[95] S. Chakravarty and P. J. Thadikaran, *Introduction to Iddq Testing*: Kluwer Academic Publishers, 1997.

[96] S. Naik, F. Agricola, and W. Maly, "Failure analysis of high-density CMOS SRAMs," *IEEE Design and Test*, pp. 13–23, June 1993.

[97] R. C. Aitken, "A comparison of defect models for fault location with Iddq measurement," in *Int. Test Conf.*, 1992, pp. 778–787.

[98] A. Gattiker and W. Maly, "Current signature: Application," in *Int. Test Conf.*, 1997, pp. 156–165.

[99] R. Arnold, T. Bode, M. Feuser, and H. U. Wedekind, "Experiences with implementation of Iddq test for identification & automotive products," in *Int. Test Conf.*, 1997, pp. 127–135.

[100] T. W. Williams, R. Kapur, and M. R. Mercer, "Iddq testing for high performance CMOS—The next ten years," in *Europ. Design and Test Conference*, 1996, pp. 578–583.

[101] M. Sachdev, "Deep sub-micron Iddq testing: Issues and solutions," in *Europ. Design and Test Conference*, 1997, pp. 271–278.

[102] P. Nigh, D. Vallett, A. Patel, and J. Wright, "Failure analysis of timing and IDDq-only failures from the SEMATECH test methods experiment," in *Int. Test Conf.*, 1998, pp. 43–52.

[103] J. T. Y. Chang, C. W. Tseng, Y. C. Chu, S. Wattal, M. Purtell, and E. J. McCluskey, "Experimental results for Iddq and VLV testing," in *IEEE VLSI Test Symp.*, 1998, pp. 118–123.

[104] J. F. Frenzel and P. N. Marinos, "Power supply current signature (PSCS) analysis: A new approach to system testing," in *Int. Test Conf.*, 1987, pp. 125–129.

[105] A. E. Gattiker and W. Maly, "Current signature," in *IEEE VLSI Test Symp.*, 1996, pp. 112–117.

[106] A. E. Gattiker and W. Maly, "Toward understanding Iddq only fails," in *Int. Test Conf.*, 1998, pp. 174–183.

[107] J. S. Beasley, A. W. Righter, C. J. Apodaca, S. P. Mozafari, and D. Huggett, "Idd pulse response testing applied to complex CMOS ICs," in *IEEE Int. Test Conf.*, 1997, pp. 32–39.

[108] Y. K. Malaiya, A. P. Jayasumana, Q. Tong, and S. M. Menon, "Enhancement of resolution in supply current based testing for large ICs," in *IEEE VLSI Test Symp.*, 1991, pp. 291–296.

[109] M. Rullan, C. Ferrer, J. Oliver, D. Mateo, and A. Rubio, "Analysis of Issq/Iddq testing implementation and circuit partitioning in CMOS cell based design," in *Eur. Design and Test Conf.*, 1996, pp. 584–588.

[110] R. Rajsuman, "Design-for-Iddq-testing for embedded cores based system-on-a-chip," in *Proc. IEEE Int. Workshop on Iddq Testing*, 1998, pp. 69–73.

[111] "Preliminary outline of IEEE P1500's scalable architecture for testing embedded cores," in *Proc. IEEE Int. Workshop on Testing Embedded Cores Based Systems*, 1998.

[112] M. Roncken, "Defect oriented testability for asynchronous ICs," *Proc.IEEE*, vol. 87, no. 2, pp. 363–375, Feb. 1999.

[113] D. P. Vallett and J. M. Soden, "Finding fault with deep submicron ICs," *IEEE Spectrum*, pp. 39–50, Oct. 1997.

[114] J. A. Kash, J. C. Tsang, R. F. Rizzolo, A. K. Patel, and A. D. Shore, "Backside optical emission diagnosis for excess Iddq," *IEEE J. Solid State Circuits*, vol. 33, no. 3, pp. 508–511, Mar. 1998.

[115] J. B. Kim, S. J. Hong, and J. Kim, "Design of a built-in current sensor for Iddq testing," *IEEE J. Solid State Circuits*, vol. 33, no. 8, pp. 1266–1272, Aug. 1998.

[116] S. Davidson, "Is Iddq yield loss inevitable," in *Int. Test Conf.*, 1994, pp. 572–579.

[117] R. Perry, "Iddq testing in CMOS digital ASICs—Putting it all together," in *Int. Test Conf.*, 1992, pp. 151–157.

[118] R. Rajsuman, *Digital Hardware Testing*. Norwood, MA: Artech House, 1992, ISBN 0-89 006-580-2, ch. 12, pp. 263–295.

[119] *Proc. Int. Technology Roadmap for Semiconductor*, Nov. 1999.

Rochit Rajsuman (S'84–M'86–SM'92) After receiving the Ph.D. degree in electrical engineering from Colorado State University, Boulder, he served on faculty in the Department of Computer Engineering and Science at Case Western Reserve University for almost seven years.

He left academia to join LSI Logic as product manager for test methodologies. In that role he productized a number of test solutions for LSI Logic including Iddq testing. From LSI Logic he moved to a media processor start-up and now works as Manager of Test Research at Advantest America R&D Center. He has authored number of patents including two on Iddq testing and published over 60 papers in refereed journal and conference. He has authored two books "Digital Hardware Testing," 1992 and first monograph on Iddq testing "Iddq Testing for CMOS VLSI," 1995, both with Artech House Inc., a division of Horizon House. He also co-edited the first book on Iddq testing "Bridging faults and Iddq Testing," IEEE Computer Society Press, 1992.

Dr. Rajsuman is a Golden Core member of the IEEE Computer Society, a member of Tau Beta Pi and Eta Kappa Nu. In 1995, he co-founded IEEE Int. Workshop on Iddq. He also co-founded two other international conferences, IEEE Int. Workshop on Memory Technology, Design and Testing, and IEEE Int. Workshop on Testing Embedded Core-based Systems; and now serves on the Steering Committee of all three workshops. He was a founding member of the IEEE P1500 Working Group to define embedded core test standards. He also serves on the technical program committees of various other conferences including International Test Conference.