IDDQ Testing

Sungho Kang
Yonsei University
Outline

- Introduction
- Defect Detection with IDDQ
- Fault Detection with IDDQ
- IDDQ Test Patterns
- IDDQ Instrumentation
- IDDQ Limit Setting and Characterization
- Design Consideration of IDDQ Testing
- Delay Faults and IDDQ Testing
- Conclusion
What is IDDQ?

• IDDQ Testing is the process of measuring IDDQ
• IDDQ is the IEEE symbol for the quiescent power supply current in an MOS circuit.
  ▪ In our cases we are referring to IDDQ in CMOS ICs
• Perfect CMOS circuits have IDDQ values typically less than 100 nA
  ▪ There is no direct conducting path between Vdd and Vss
• Most CMOS IC defects elevate IDDQ several orders of magnitude greater than non-defective circuit
• IDDQ Testing is the most sensitive way to detect the majority of CMOS defects
Limitation of Logic Testing

- Gate to source short in N1 of a NAND gate
- Voltage testing based on stuck-at fault model cannot fully detect transistor level defects (about 40%)
IDDQ : Quiescent Current

- When CMOS is not actually switching, one transistor in CMOS pair is always off
- $I_{ddq}$ refers to the quiescent power supply current drawn by CMOS circuits in stable states
- Draws only a leakage current on the order of nA
- $I_{ddq}$ can become as high as several mA
High IDDQ Indicates a Defective IC

- When CMOS is not actually switching, one transistor in CMOS pair is always off
- Draws only a leakage current on the order of nA
- IDDQ can become as high as several mA
Why IDDQ Testing?

- Chips with transistor leakage faults may pass even 100% fault graded functional production tests
- Reliability studies involving life test
  - Burn-in testing
  - Leakage faults could develop into hard stuck-at faults
  - Source of long time failure
- As intensity and complexity of VLSI are increased
  - Reliability becomes very important
  - Aerospace, Automobile
Advantages of IDDQ Testing

- **Direct Observability**
  - \( \gg 50\% \) of transistors tested with only a few vectors

- **Detection of defects that do not cause functional failure**
  - Identification of subtle defects and failure mechanisms in addition to those that affect logic functions

- **Greatly increased detection of common physical defects**
  - Gate oxide shorts, Interconnect shorts, Interconnect opens

- **100% coverage of stuck-at faults for many designs**

- **Reduced test vector count**

- **Simplified test vector generation and fault simulation**

- **No additional on-chip circuitry required**
## Test Method Success for Defects

- **P-poor; F-fair; G-good**

<table>
<thead>
<tr>
<th>DEFECT</th>
<th>FUNCTIONAL</th>
<th>STUCK-AT</th>
<th>STUCK-OPEN</th>
<th>$I_{DDQ}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gate shorts</td>
<td>P</td>
<td>P</td>
<td>P</td>
<td>G</td>
</tr>
<tr>
<td>Bridges</td>
<td>F</td>
<td>F-G</td>
<td>G-F</td>
<td>G</td>
</tr>
<tr>
<td>Parastics &amp; p-n leakage</td>
<td>P</td>
<td>P</td>
<td>P</td>
<td>G</td>
</tr>
<tr>
<td>Punch through</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>G</td>
</tr>
<tr>
<td>Open drain</td>
<td>P</td>
<td>F</td>
<td>G</td>
<td>F</td>
</tr>
<tr>
<td>Open gate</td>
<td>F</td>
<td>G</td>
<td>G</td>
<td>F</td>
</tr>
<tr>
<td>Open t-gate</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>G</td>
</tr>
</tbody>
</table>
Recent IDDQ Achievements

- >50% reduction in IC production cost due to improved screening of defects at wafer test/sort
- Major assembly line and customer field quality improvement after implementation for scan ASICs with >99% stuck at fault coverage
- >50% drop in burn-in rejects and resultant ability to reduce burn-in
Types of CMOS Defects Mechanisms

- **Shorts**
  - Electrical connections between circuit nodes that are not usually zero ohms
- **Opens**
  - Breaks in the interconnect materials
- **Degradations**
  - Defects mechanisms that may alter circuit performance and may produce shorts or opens
- **IDDQ Leakage**
  - Can include all of the above
Memory Testing with IDDQ

- Gate oxide shorts are a strong concern in memories
  - A large gate oxide area exists
- IDDQ testing in memories is efficient for many types of defects; gate shorts, stuck cells, or bridged nodes
- An IDDQ measurement of the checkerboard and inverse checkerboard patterns provides rapid coverage of the majority of transistors (but not all)
- A memory node-state test is relatively short, $O(4n)$
- IDDQ measured at the checkerboard and inverse checkerboard patterns will detect all gate shorts in the four-transistor cross-coupled effort
- To detect gate shorts in the write access transistors, IDDQ must be measured at each phase of the write signal
- IDDQ is very powerful for detecting memory defects in CMOS SRAM’s with a minimum of effort
IDDQ and CMOS Defect Types

- All short circuit defects are most efficiently and exactly detected by the IDDQ test measurement.
- It is necessary that a test vector activate the defect (controllability) and IDDQ measurement be taken.
- Functional or SAF test sets will not defect many of these defects except by chance since the primary effect of these short defects can be either weakening of logic voltages or a bridge masking effect.
- None of the open circuit defects are guaranteed detectable by the IDDQ test, but neither are functional or stuck-at fault test sets.
  - The IDDQ test increases the probability of open circuit defect detection with the advantage of a small test set.
- Certain defects do not elevate IDDQ nor are they efficiently detected by a functional type test set.
ICs That Have High IDDQ

- Reliability concerns
  - These ICs often have lower reliability (early functional failure)
  - High current states may cause premature battery failure

- Quality/Yield concerns
  - Current may be symptomatic of a significant problem
  - Causal defects/mechanisms may worsen in time
  - Ignoring defects produces
    - Inaccurate quality/performance measurements/metrics
    - Incorrect test/experiment conclusions
Fault Detection with IDDQ

• Several Faults can be used as test metrics
  ▪ Stuck-at faults
  ▪ Bridging faults
  ▪ Stuck-open faults
  ▪ Stuck-on faults
  ▪ Delay faults

• Faults are considered because
  ▪ Many test standards evolved from faults not defects
  ▪ Computer simulation programs typically use faults
  ▪ Legal and vendor/customer requirements often specify fault coverage
SAF Detection with IDDQ Testing

- SAFs are defined for CMOS ICs as logic failures that occur when a logic gate input or output node is bridged to one of the power supply rails with a zero ohm connection.
- SAFs are time invariant and therefore cause failures at any frequency of operation.
- SAFs have been the backbone of DFT throughout the 1970-80 era and despite advances in defect knowledge, the SAF is still a required metric by many customers.
- A significant observation was that SAF coverage can be obtained with IDDQ testing using test vector sizes that may be 1% of conventional vectors.
Detection of Redundant Faults

- A B C D E F E s-a-1 detection
- 0 0 0 0 0 0 Yes
- 1 1 1 1 1 1 No

![Fault Detection Diagram]
Comments on SAF Testing

- The SAF is a poor metric for CMOS circuits, but it lingers in the industry
- The IDDQ test easily satisfies most SAF requirements
- Previously untestable circuits, such as typical microprocessors can now be evaluated for those customers who demand it
- Many who try the IDDQ test technique for SAF coverage have discovered the additional advantages of IDDQ testing such as
  - 100% gate oxide short coverage
  - Better bridge and delay defect coverage
  - Detection of some open circuit defects
  - Detection of certain design properties
  - Detection of certain process problems
Detection of Bridging Faults

- Bridging faults and bridging defects are similar except that bridging faults assume a zero ohm defect, but bridging defects assume that any impedance is possible.
- Logic (voltage) detection problems are the same as those for other types of defects:
  - Identification of plausible node bridges
  - Generation of a test vector that activates the defect (controllability)
  - Setting of observation paths to observe logic response
- IDDQ testing eliminates the 3rd requirement and greatly simplifies the test.
## Logic vs Current Testing

### Fault Detection

#### Bridging faults detected with voltage sensing

<table>
<thead>
<tr>
<th>ISCAS85</th>
<th>Patterns</th>
<th>Faults</th>
<th>Missed</th>
<th>%detected</th>
</tr>
</thead>
<tbody>
<tr>
<td>c17</td>
<td>6</td>
<td>74</td>
<td>1</td>
<td>98.7%</td>
</tr>
<tr>
<td>c432</td>
<td>53</td>
<td>3968</td>
<td>174</td>
<td>95.6%</td>
</tr>
<tr>
<td>c499</td>
<td>58</td>
<td>7806</td>
<td>362</td>
<td>95.4%</td>
</tr>
<tr>
<td>c880</td>
<td>67</td>
<td>7909</td>
<td>99</td>
<td>98.8%</td>
</tr>
<tr>
<td>c1355</td>
<td>87</td>
<td>12103</td>
<td>62</td>
<td>99.5%</td>
</tr>
<tr>
<td>c1908</td>
<td>108</td>
<td>14203</td>
<td>338</td>
<td>97.6%</td>
</tr>
</tbody>
</table>

#### Bridging faults detected with current sensing

<table>
<thead>
<tr>
<th>ISCAS85</th>
<th>Patterns</th>
<th>Faults</th>
<th>Missed</th>
<th>%detected</th>
</tr>
</thead>
<tbody>
<tr>
<td>c17</td>
<td>5</td>
<td>75</td>
<td>0</td>
<td>100.0%</td>
</tr>
<tr>
<td>c432</td>
<td>40</td>
<td>4086</td>
<td>56</td>
<td>99.59%</td>
</tr>
<tr>
<td>c499</td>
<td>99</td>
<td>8167</td>
<td>1</td>
<td>99.99%</td>
</tr>
<tr>
<td>c880</td>
<td>27</td>
<td>8006</td>
<td>2</td>
<td>99.97%</td>
</tr>
</tbody>
</table>
Detection of CMOS Stuck-Open

- CMOS stuck-open faults shows that IDDQ has a significant probability of detecting CMOS stuck-open faults due to:
  - Transistor contention caused by the memory effect of the SOF
  - Floating nodes caused by the SOF turn on CMOS transistor pairs
- Consider SOFs as high likelihood non-target faults of IDDQ testing
IDDQ Testing and Delay Faults

- Delay faults denote the collection of all defects that cause a circuit fail at a frequency less than that of a defect-free circuit.
- Delay faults will pass functional tests at DC.
- Delay faults are an increasing concern as we build products whose clock drops from the 100ns period to the 5-10ns period.
- What does the relatively slow speed IDDQ test have to do with the detection of delay faults?
  - A large percentage of defects that cause delay faults are sensitive to detection by the IDDQ test.
IDDQ Test Patterns

- IDDQ testing refers to the measurement of quiescent power supply current in CMOS circuits to cover many types of defects
- An IDDQ test pattern set consists of test vectors used for IDDQ measurement
- If the CUT contains defects targeted by an IDDQ test vector, the CUT draws excessive quiescent current for the test vector
- IDDQ test patterns are also used for defect diagnosis and can be lead to specific defective devices in the IC
Requirements for IDDQ Test Patterns

- A test vector must create one or more low resistance paths from Vdd to Vss in the presence of target defect(s)
- The acceptable number of IDDQ test vector is several orders of magnitude less than what may be reasonable for voltage based logic testing
Obtaining IDDQ Test Patterns

- Use the entire functional or structural test pattern and measure IDDQ for each test vector
  - Every Vector IDDQ Test Patterns
- Select a subset of the functional or structural test pattern and measure IDDQ for selected test vectors
  - Selective IDDQ Test Patterns
- Generate test patterns specifically for IDDQ testing to supplement the traditional functional or structural testing
  - Supplemental IDDQ Test Patterns
Every Vector IDDQ Test Patterns

- Functional or structural test patterns are used
  - the only modification necessary is addition of control statements to slow down the test application rate

- Advantages
  - No additional test generation effort or software is required
  - No fault model is necessary; coverage for all defects that are sensitized and to which IDDQ is sensitive are covered
  - Effective for both combinational and sequential circuits

- Disadvantages
  - Coverage of defects for which IDDQ testing is effective is constrained by the quality of functional or structural test patterns
  - Defect coverage is not quantified
  - For dynamic circuits, it may not be allowable to slow down testing during the application of at least some of the vectors
  - Circuit is not tested at system speed
  - The number of test vectors is usually too large for the methodology to be practical in large volume production
Selecting IDDQ Test Patterns

- A straightforward procedure could be
  - Use logic simulation to capture the logic values at the terminals of each transistor upon the application of every test vector
  - Determine which leakage faults would be caught by IDDQ measurement at each test vector
  - Select a small subset of test vectors to cover as many leakage faults as possible

- Practical problems
  - The models used for logic simulation may not all be at transistor level
  - Transistor level simulation of VLSI circuits may be beyond the capacity (memory and speed) of logic simulators
  - Even if transistor level simulation is feasible, the amount of data required to be captured and analyzed will be beyond reasonable limits
Selective IDDQ Test Patterns

• A subset of the functional or structural test pattern set is selected for IDDQ testing of modeled faults

• Advantages
  ▪ Small number of test vectors (often under 1%) can be selected without loss of coverage; methodology is practical for large volume production IDDQ testing
  ▪ Fault coverage is quantified
  ▪ Effective for both combinational and sequential circuits
  ▪ Circuits can be tested at system speed except for selected vectors
  ▪ Dynamic circuits can be accommodated

• Disadvantages
  ▪ Fault model is necessary; coverage shows IDDQ test effectiveness for only those defects that have a representation in the fault model
  ▪ Coverage is constrained by the quality of functional or structural logic test pattern
  ▪ Software for test vector selection is required
Supplemental IDDQ Test Patterns

- Test vectors are generated specifically for IDDQ testing
  - logic testing is done at full system speed and is supplemented by IDDQ testing for the generated pattern
- Fault models must be used to make the process of IDDQ test vector generation feasible and appropriate software must be developed
- Bollinger et al (ITC 91) have developed a methodology that supports IDDQ test generation for bridging faults
- Chen et al (ITC 91) have proposed an approach for generating tests for switch level circuits using both IDDQ and logic test generation; node stuck-at, transistor stuck-open and transistor stuck-on faults are considered
Supplemental IDDQ Test Patterns

- Test vectors are generated specifically for IDDQ testing of modeled faults

- Advantages
  - Small number of effective test vectors can be generated to supplement logic testing
  - May also be employed to supplement Selective IDDQ test patterns
  - Fault coverage is quantified
  - Coverage is independent of the quality of logic test pattern
  - Logic testing can be done at system speed

- Disadvantages
  - Fault model is necessary; coverage shows effectiveness for only those defects that have a representation in the fault model
  - Practicality of test generation methods for large sequential circuits is yet to be established
  - Software for test vector generation is required
Off-Chip Measurement

- The diode resistor method has been replaced with other techniques for new ICs.
- The FET-resistor method was used for several years but its limit is around 10 uA.
- An OP amp circuit has been evaluated and also has a limit around 10-15 uA.
- The bit-current option has a sensitivity to about 0.5uA; its measurement rate at 1 uA is about 10-15 KHz.
- The Keating circuit is sensitive (pA-uA range), but as with all circuits runs slow for low current measurements; at 1 uA, the measurement rate may be about 5 KHz.
Off-Chip Measurement

Instrumentation

\[ V_{DD} \rightarrow I_{DD} \rightarrow \text{DUT} \]

\[ R_m \quad V_{DD} \quad V_O \]

\[ V_O \quad I_{DDQ} \quad \text{Similarly, for} \quad i_{DD} \]

No Defect

\[ \text{TIME} \]

Defect

\[ \text{TIME} \]

No Defect

\[ \text{TIME} \]
On-Chip Circuit

- High speed testing
- Detection of small currents
- Use of conventional test equipments
- Easy to test

- Area overhead
- Performance degrade
  - Output voltage
  - Operation speed
- Partition required
Summary

- IDDQ instrumentation down to the 10-15 uA range is quite easy to install using the FET/resistor or OP amp circuits.
- IDDQ limits on the order of 1uA are slightly more difficult; the keating floating Vdd node circuit looks good or can use very sensitive instrumentation with a measurement resistor.
- All instrumentation methods have rate vs sensitivity tradeoffs:
  - at IDDQ = 1uA, f_{test} = 3-5kHz
  - at IDDQ = 10uA, f_{test} = 50kHz
- On-chip circuits will probably grow in development; higher rates and more sensitivity are reported.
IDDQ Limit Setting

- The 1mA limit setting is very coarse, catching gross leakage defects and CMOS stuck-at faults
  - It is better than nothing but perhaps too much
- Some have used the RMS current of the power supply
  - This provides an indication of power consumption with frequency, but not much about random defects
- The 20-100 uA range is easily obtainable with rapidly assembled IDDQ instrumentation
- The defect detection is still coarse in this range, but you get many bridges, gate shorts, stuck-opens, etc.
- There are probably many fab houses that put out circuit quality in this range
Reliability

• SANDIA data specifically show that gate shorts are a significant reliability risk
  ▪ The part may leak, show functionality, but subsequently fail
• The Ford data show that 21-51% of ICs with high IDDQ pre-burn-in failed functional tests after burn-in
• Some leakage mechanisms might not cause reliability failures, but their presence masks the ability to detect those leaky defects that do cause failure
Ideal Design for IDDQ Testability

- Fully CMOS Design
- 100% static
  - Nodes always driven to Vdd/Vss except when switching
  - No minimum clock frequency
- 100% complementary
  - Nodes driven by complementary P/N networks

The ideal design for IDDQ testing is a fully-static, fully-complementary CMOS IC

IDDQ testing can be performed on some ICs that are not full CMOS designs
Conclusion

- Most CMOS IC defects elevate IDDQ several orders of magnitude greater than non-defective circuit.
- IDDQ Testing is the most sensitive way to detect the majority of CMOS defects:
  - Direct Observability
  - Greatly increased detection of common physical defects
  - 100% coverage of stuck-at faults for many designs
  - Reduced test vector count
  - Simplified test vector generation and fault simulation
  - No additional on-chip circuitry required
- Need research on current sensing and test set compaction.