

# Stuck-At Fault: A Fault Model for the next Millennium?

Janak H. Patel

Department of Electrical and Computer Engineering  
University of Illinois at Urbana-Champaign

© 2005 Janak H. Patel

# Stuck-At Fault

“I tell you, I get no respect!”

-Rodney Dangerfield, Comedian

“The news of my death are highly exaggerated”

-Mark Twain, Author

## Stuck-At Fault a Defect Model?

You can call it -

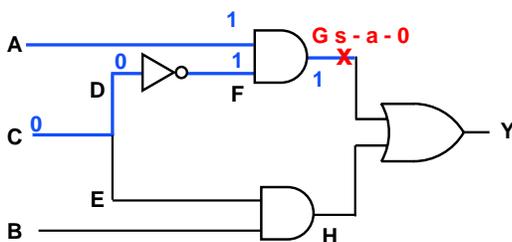
- Abstract
- Logical
- Boolean
- Functional
- Symbolic
- or Behavioral ..... Fault Model

But don't call it a Defect Model!

## Stuck-At Fault as a Logic Fault

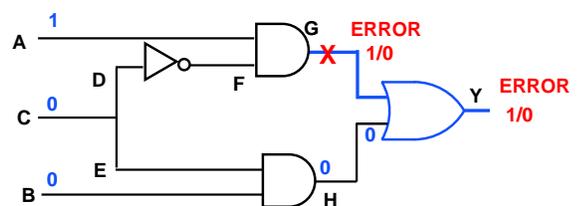
- Stuck-at Fault is a **Functional Fault** on a Boolean (Logic) Function Implementation
- It is not a Physical Defect Model
  - Stuck-at 1 does not mean line is shorted to  $V_{DD}$
  - Stuck-at 0 does not mean line is grounded!
- It is an abstract fault model
  - A logic stuck-at 1 means when the line is applied a logic 0, it produces a **logical error**
  - A logic error means 0 becomes 1 or vice versa

## Fault Excitation



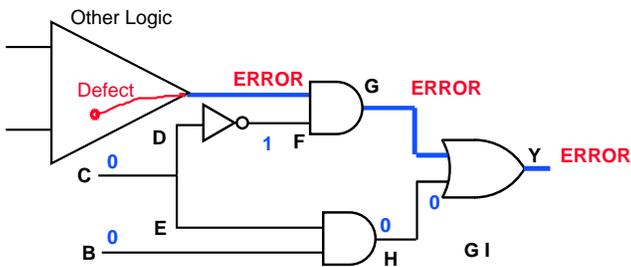
Activates the fault s-a-0 on line G by applying a logic value 1 in line G

## Propagate Error To Primary Output Y



Test Vector A,B,C = 1,0,0 detects fault G s-a-0

## Unmodeled Defect Detection



7

## Defect Sites

- **Internal to a Logic Gate or Cell**
  - Transistor Defects – Stuck-On, Stuck-Open, Leakage, Shorts between terminals
- **External to a Gate or Cell**
  - Interconnect Defects – Shorts and Opens

8

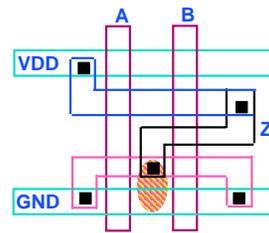
## Defects in Physical Cells

- Physical Cells such as NAND, NOR, XNOR, AOI, OAI, MUX2, etc.
  - For primitive gates such as NOT, NAND and NOR, stuck-at tests are derived for faults on the pins.
  - For complex cells such as XOR, XNOR, AOI, OAI, and MUX2 etc, Stuck-at Tests are assumed to be derived on faults on gate equivalent models.
  - How good are these test vectors for a variety of defects?
- Do we need additional vectors?
  - Do we need transistor level details?

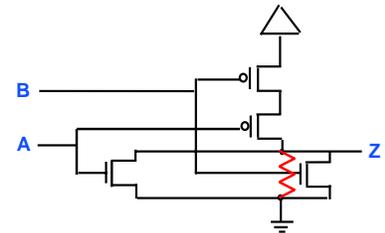
9

## Fault Modeling

### ● Physical



### ● Electrical

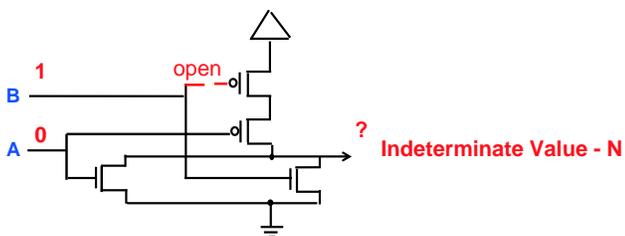


### ● Logical



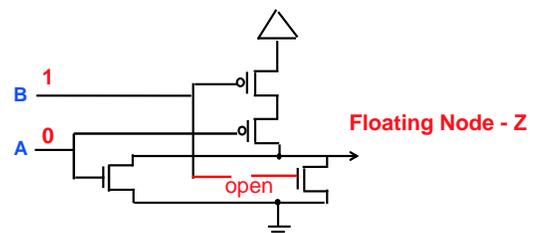
10

## Non-Logical Values



11

## Non-Logical Values



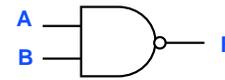
12

## Defect Detection in a NAND Gate

- For a 2-input NAND gate, the complete stuck-at test set is: AB = 01, 10 and 11
- With a defect in the NAND cell, the gate may produce any combinations of 0, 1, N, Z
  - N is an indeterminate logic value (active, driven)
  - Z is a floating node with unknown charge (passive)
- Each of 4 possible input vectors can produce any of the 4 possible output values
  - 256 possible defective behaviors for 2-input NAND
    - Infinitely many delay and current behaviors
- Is the stuck-at test set 01, 10 and 11 sufficient?

13

## Two-Input NAND Gate



AB	F	a/0	b/0	a/1	b/1	F5	F6	F7	F8	F9	...	F256
00	1	1	1	1	1	1	1	Z	...	0	...	Z
01	1	1	1	0	1	Z	1	0		1		Z
10	1	1	1	1	0	1	1	Z		1		Z
11	0	1	1	1	1	0	N	0		0		Z

Fault Dictionary

14

## Defect Characterization

- Inductive Contamination Analysis (ICA)**  
*J. Khare, W. Malay and N. Tiday, VLSI Test Symp. 1996, pp. 407-413*  
 "Inductive Fault Analysis (IFA) is inadequate for three dimensional defects in multi-layer cells"
  - Experiment on 2-input NAND cell with 1000 particle contamination simulations. Assumed 84 major process steps, 2-metal C-MOS.
  - Reported 22 different fault behaviors in the paper
- Stuck-at test set (01,10,11) was "sufficient" for all behaviors** (*my interpretation not theirs*)

15

## Vector 00 and 2-input NAND

### Pseudo Theorem:

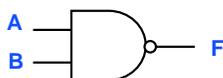
In a 2-input NAND CMOS cell, there does not exist a real physical defect that requires test vector 00 for its exposure.

### Proof:

If such a defect existed, it would make the gate "more functional than a NAND gate."

16

## Two-Input NAND Gate



AB	F	a/0	b/0	a/1	b/1	F5	F6	F7	F8	$A \oplus B$	F9
00	1	1	1	1	1	1	1	Z	...	0	0
01	1	1	1	0	1	Z	1	0		1	1
10	1	1	1	1	0	1	1	Z		1	1
11	0	1	1	1	1	0	N	0		0	0

Fault Dictionary

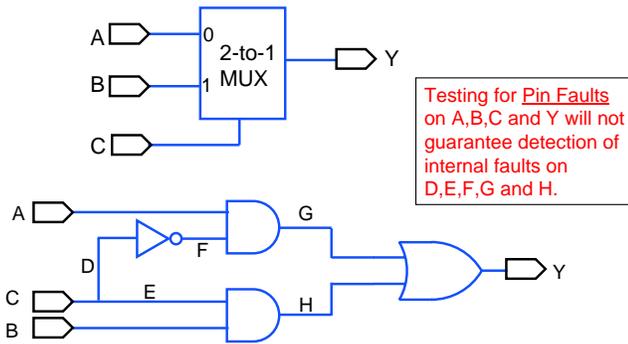
17

## Stuck-at tests for other Cells

- It can be shown that for all simple gates and complex gates with fan-out free logic, stuck-at test for pin faults is sufficient to expose any defect inside the Cell
  - Simple Gates, NAND, NOR, NOT
  - Complex fan-out free Gates, AOI, OAI
    - Even functions such as:  $[(A+B)(C+D)+E]INVERT$
- Pin fault stuck-at tests are not adequate for complex gates with internal fanout-reconvergence
  - XOR, XNOR, MUX

18

## Multiplexer Expansion



19

## What about N and Z values?

- Some of the defects within a cell produce indeterminate logic N or floating node Z values for some of the stuck-at vectors.
  - If a clean logic error (0-to-1, or 1-to-0) is possible, a stuck-at test vector will expose it.
  - That means N and Z values cannot be avoided by using test vectors other than stuck-at vectors.
  - If a clean logic error is not possible, Stuck-at vectors are sufficient to “expose” the defect by changing a correct logic value to either N or Z.

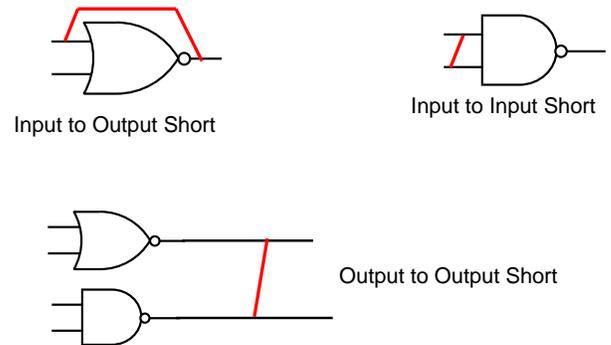
20

## Defects External to a Gate

- Opens
  - All opens external to a gate are detected by a stuck-at fault test set
- Shorts
  - Input-to-Input Shorts on the same Gate
  - Input-to-Output Shorts on the same Gate
  - Output-to-Output Shorts on different Gates

21

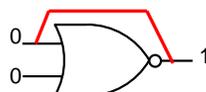
## Shorts



22

## Input to Output Short

- In a simple CMOS gate, if the short causes an Error then Input value is forced upon the output - Vierhaus, Meyer and Glaser, ITC-93
- This is also true for complex CMOS gates such as And-Or-Invert (AOI) and Or-And-Invert (OAI) - Cusey, M.S. Thesis, Illinois 1993
- Test Vectors for Input and Output Stuck-at Faults cover Input-to-Output Shorts
  - Experiments Confirm this



23

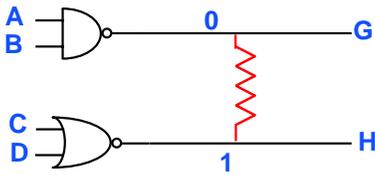
## Input-to-Output Bridging Faults

- Validation of Stuck-at Fault Test Set Coverage
  - Circuits with Complex Gates
  - All Bridging Faults on the Input and Output on the same Gate
  - Fault Simulation with E-PROOFS

Circuit	Vectors	Total Faults	% Faults Detected		
			0k $\Omega$	1k $\Omega$	2k $\Omega$
C432	100	371	100%	100%	43%
C499	190	274	100%	100%	97%
C880	128	336	94%	90%	70%

24

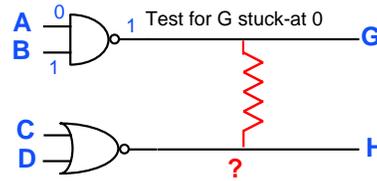
## Logic Model for a Short



FAULT-FREE	FAULTY	FAULT MODEL
$G, H = 0, 1$	$G, H = 0, 0$ or $G, H = 1, 1$	$H$ s-a-0 when $G=0$ $G$ s-a-1 when $H=1$
$G, H = 1, 0$	$G, H = 0, 0$ or $G, H = 1, 1$	$G$ s-a-0 when $H=0$ $H$ s-a-1 when $G=1$

25

## Shorts and Stuck-at Tests



- Assume H dominates G with the Bridge present
- Test for G stuck-at 0 has no control over node H
- Probability that H has the correct logic value to excite the Bridge is 0.5

26

## Probability of Detection

- The four stuck-at faults on nodes G and H require a minimum of two test vectors
- Each test vector has a probability of Bridge Excitation of 1/2
  - Probability that two test vectors miss the excitation of the bridge is 1/4
  - Lower bound on expected bridge coverage is 75%
  - For most stuck-at test sets, a node gets tested many more times than 2

27

## Repeated Detections

- Example - ISCAS89 fullscan circuit S38417
  - 99 test vectors, 31,015 faults detected

Number of Repetitions	Number of Faults
1	3,411
2	1,710
3	1,262
4	1,043
5	861
6	925
7	821
8	834
9	808
>10	19,340

28

## Probability of Bridge Detection

- Assume the stuck-at test set detects each node n times
  - Probability of Bridge being detected is  $(1 - 1/2^n)$
- For example, let us say each stuck-at fault gets detected 5 times.
  - That means each node gets detected 10 times.
  - The probability of Bridge detection is 99.9%
- Caveat: The Bridge must cause a Detectable error.
  - High Resistance Bridges do not affect the logic value, and hence are undetectable by a static logic test.

29

## Bridge Coverage with SSF Test Set

- Simulation of extracted bridges with Stuck-at Test Sets using very accurate electrical level simulator (EPROOFS, Greenstein, Patel, ICCAD 1992) shows a very high coverage in ISCAS circuits.
- Output to Output Bridges have comparable coverage to a stuck-at fault coverage
- Input to Input Bridges have lower coverage because many of them are logically redundant

30

## SSF Test Set and Bridging Faults

Circuit	Stuck-at Vectors	Output to Output Bridges		
		0k $\Omega$	1k $\Omega$	2k $\Omega$
C499	184	99.8%	77.5%	1.8%
C880	128	96.9	46.0	3.9
C1908	138	98.8	71.6	1.8

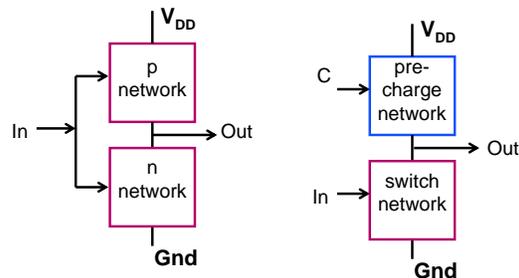
Circuit	Stuck-at Vectors	Input to Input Bridges		
		0k $\Omega$	1k $\Omega$	2k $\Omega$
C499	184	91.5%	84.1%	0.0%
C880	128	52.7	43.9	0.0
C1908	138	87.9	56.6	0.0

(source: J. Cusey and J. Patel, ITC 1997, pp 838-847)

31

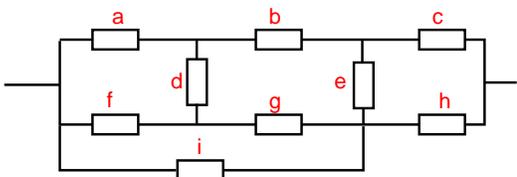
## Large Custom Blocks

- Two major classes
  - Fully Complementary CMOS networks
  - All Others "One-Sided" networks.



32

## Modeling Switch Networks



Express the network as AND-OR or OR-AND Network  
 C. E. Shannon, "A symbolic analysis of relay and switching networks"  
 Trans. AIEE, 1938.

**AND-OR:** Each series path corresponds to an AND

$abc + abeh + adgec + adgh + fdhc + fdbeh + fgec + fgh + iec + ih$

**OR-AND:** Each cut-set corresponds to an OR

$(a+f+i)(a+d+g+i).....$

Generate a stuck-at test set on either network

33

## Fault Coverage Metrics

- **A Stuck-at faults on Transistor terminals?**
  - Not a very meaningful measure because Logic 1 and 0 do not always occur on all terminals, **N and Z values abound!**
    - ◆ Gives unnecessarily pessimistic coverage
  - Complementary transistors cannot be independently controlled, results in many untestable faults.
    - ◆ Gives unnecessarily pessimistic coverage
- **Gate level logic stuck-at coverage is already hard! Don't make it any harder!**

34

## Final Thoughts

- **Logic Stuck-at Fault**
  - Good for defects within a cell
    - ◆ Any lower level model is too complex and inaccurate
  - Good for defects outside of a cell
    - ◆ Bridges easy to cover without explicit targeting
  - Easy to model custom blocks for an ATG
  - Coverage metrics are well-defined
  - Automation is well-understood

"Things should be made as simple as possible, but not any simpler" *Albert Einstein*

35