Digital System Testing

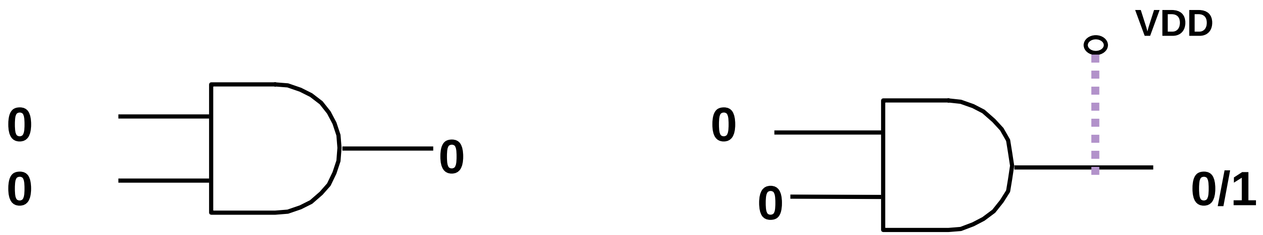
Ref: Digital System Testing & Testable Design by Miron Abramovici

Testing & Diagnosis

- What is testing?
  - The system is exercised and its resulting response is analyzed to ascertain whether it behaved correctly
- What is diagnosis?
  - Locate the cause of the misbehavior of the system

Basic Concept of Testing

Testing: To tell whether a circuit is good or bad



- **Verification:** To verify the correctness of a design
- **Diagnosis:** To find the faulty site
- **Reliability:** To find whether a good system will work correctly or not after some time
- **Debug:** To find the faulty site and try to eliminate the fault

Significance of Testing?

- Economics!
  - Reduce test cost (enhance profit)
    - Automatic test equipment (ATE) is extremely expensive
  - Shorten time-to-market
    - Market dominating or sharing
  - Guarantee IC quality and reliability

Rule of Ten:

Cost to detect faulty IC increases by an order of magnitude

Defects detected in	Cost
Wafer	0.01 – 0.1
Packaged chip	0.1 – 1
Board	1 – 10
System	10 – 100
Field	100 – 1000

Importance of Testing

- Moore’s Law results from decreasing feature size (dimensions)
  - from 10s of μm to 10s of nm for transistors and interconnecting wires
- Operating frequencies have increased from 100KHz to several GHz
- Decreasing feature size increases probability of defects during manufacturing process
  - A single faulty transistor or wire results in faulty IC
  - Testing required to guarantee fault-free products

Importance of testing

N = # transistors in a chip

p = Prob. (a transistor is faulty)

P_f = Prob. (the chip is faulty)

$$\Longrightarrow \quad P_f = 1- (1- p)^N$$

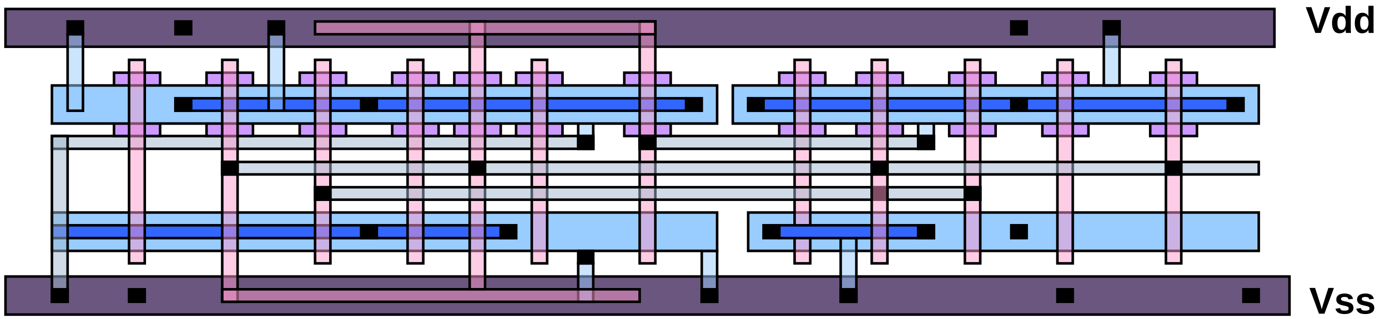
If $p = 10^{-6}$

$N = 10^6$

$$\Longrightarrow \quad P_f = 63.2\%$$

Difficulties in Testing

- **Fault may occur anytime**
  - Design
  - Process
  - Package
  - Field
- **Fault may occur at any place**



Verification Vs Testing

- Verifies correctness of design.
- Performed by simulation, hardware emulation, or formal methods.
- Performed once prior to manufacturing.
- Responsible for quality of design.

- Verifies correctness of manufactured hardware.
- Two-part process:
  - Test generation: software process executed once during design
  - Test application: electrical tests applied to hardware
- Test application performed on every manufactured device.
- Responsible for quality of devices.

Problems of Ideal Test

- Ideal tests detect all defects produced in the manufacturing process.
- Ideal tests pass all functionally good devices.
- Very large numbers and varieties of possible defects need to be tested.

Real Test

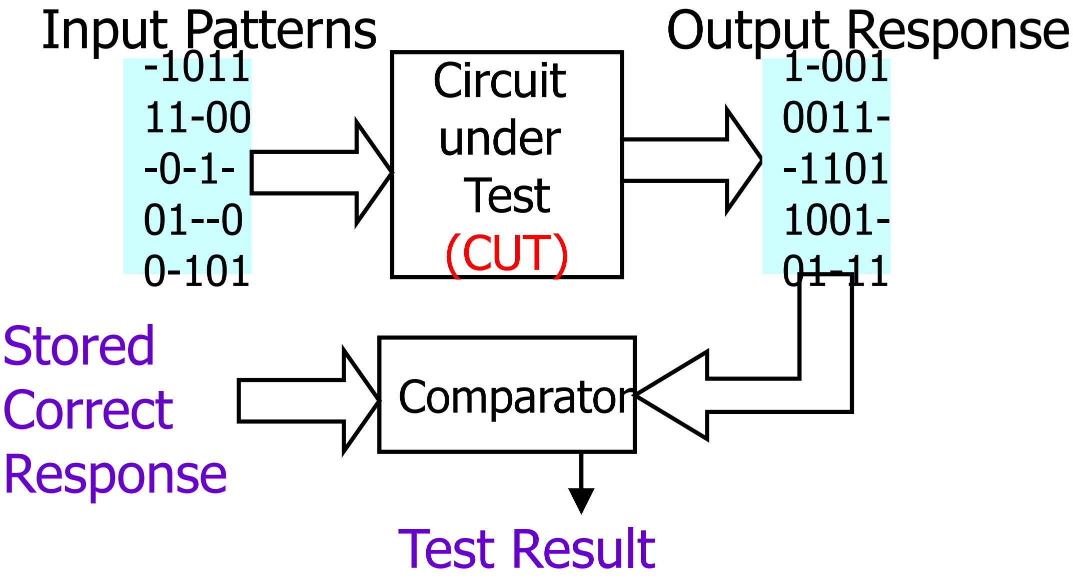
- Based on analyzable fault models, which may not map on real defects.

- Incomplete coverage of modeled faults due to high complexity.

- Some good chips are rejected. The fraction (or percentage) of such chips is called the *yield loss*.

- Some bad chips pass tests. The fraction (or percentage) of bad chips among all passing chips is called the *defect level*.

Principle of Testing



Input Patterns

-1011
11-00
-0-1-
01--0
0-101

Circuit under Test (CUT)

Output Response

1-001
0011-
-1101
1001-
01-11

Stored Correct Response

Comparator

Test Result

- Testing typically consists of
  - Applying set of test stimuli (input patterns, test vectors) to inputs of *circuit under test* (CUT), and
  - Analyzing output responses
- **The quality of the tested circuits will depend upon the thoroughness of the test vectors**

Testing - levels of abstraction

<table>
<tr><th>Control</th><th>Data</th><th>Level of abstraction</th></tr>
<tr><td colspan="2">Logic values
(or sequence of logic values)</td><td>Logic level</td></tr>
<tr><td>Logic values</td><td>Words</td><td>Register level</td></tr>
<tr><td>Instructions</td><td>Words</td><td>Instruction set level</td></tr>
<tr><td>Programs</td><td>Data structures</td><td>Processor level</td></tr>
<tr><td colspan="2">Messages</td><td>System level</td></tr>
</table>

Testing Stages

- Circuit and physical design
- Device processing (wafer-level testing)
- Device packaging (package-level testing)
- Device mounted to the circuit board (board-level testing)
- Circuit board insulated in the system (system-level testing)
- Product delivery (field-level testing)

Errors

- An instance of an incorrect operation of the system being tested.
- The error may be
  - Design error
  - Fabrication error
  - Fabrication defect
  - Physical failure
  - Environmental failure

Faults

- Fab. Error, fab. Defects & physical failures are collectively referred as physical fault.
  - Permanent
  - Intermittent
  - Transient
- Fault is detected by observing an error caused by it.
- The basic assumptions regarding the nature of logical faults are referred to as ***fault model***

Test Evaluation

- Determining the Effectiveness or Quality of a test.
- It is in the context of a fault model.
- Ratio between the no. of faults detected by the total no of faults in the assumed fault universe - ***Fault coverage*.**
- It is carried out via a simulated testing experiment called ***fault simulation*.**

Types of Testing

Criterion	Attribute of testing methods	Terminology
When is the test is performed?	□Concurrently with the normal system operation □As a separate activity	□On-line testing □Concurrent testing □Off-line testing
Where is the source of stimuli?	□Within the system itself □Applied by the an external device	□Self-testing □External testing
What do we test for?	□Design error □Fabrication errors □Fabrication defects □Infancy physical failures □Physical failures	□Design verification testing □Acceptance testing □Burn-in □Quality-assurance testing □Field testing □Maintenance testing

Types of Testing – Contd…

Criterion	Attribute of testing methods	Terminology
What is the physical object being tested?	□IC □Board □System	□Component-level testing □Board-level testing □System-level testing
How are the stimuli and/or the expected response produced?	□Retrieved from the storage □Generated during testing	□Stored-pattern testing □Algorithmic testing □Comparison testing
How are the stimuli applied?	□In a fixed (predetermined) order □Depending on the result obtained so far	□Adaptive testing

Types of Testing – Contd…

Criterion	Attribute of testing methods	Terminology
How fast are the stimuli applied?	□Much slower than the normal operation speed? □At the normal operation speed	□DC (static) testing □AC testing □At-speed testing
What are the observed results?	□The entire output patterns □Some functions of the output patterns	□Compact testing
What lines are accessible for testing?	□Only the I/O lines □I/O and internal lines	□Edge-pin testing □Guided-probe testing □bed-of-nails testing □Electron-beam testing □In-circuit testing □In-circuit emulation

Types of Testing – Contd…

Criterion	Attribute of testing methods	Terminology
Who checks the results?	□The system itself □An external device (tester)	□Self-testing □Self-checking □External testing

Fault Modeling

Logical Fault Models

- Logical faults represents the effect of physical faults on the behavior of the modeled system
  - Faults that affects the logic function
  - Delay faults

Logical Fault Models

- What do we gain by modeling physical faults as a logical faults?
- First
  - Fault analysis become logical
  - Complexity is greatly reduces
- Second
  - Some logical faults are technology independent, same faults model is applicable to many technologies.
  - Testing & diagnosis method developed for such faults remain valid despite changes in technology
- Third
  - Test derived for the logical faults can be used for physical faults, whose effect on circuit behavior is not completely known.

Explicit & Implicit Faults

- Explicit fault
  - A fault universe in which every fault is individually identified and hence the faults to be analyzed can be explicitly enumerated.

- Implicit fault
  - A fault universe by collectively identifying the faults of interest – typically by defining their characteristic properties

Structural & Functional Faults

- Structural fault
  - In conjunction with structural model
- Functional fault
  - In conjunction with functional model

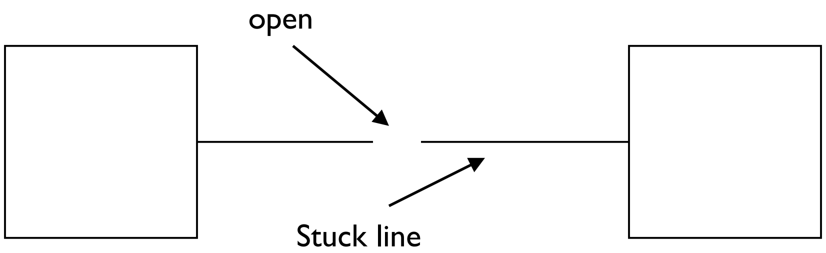
Frequent Testing Strategy

- Single fault assumption
- Many physical faults can affect an area containing several components
- Newly manufactured system prior to testing, multiple faults are likely to exist.
- Multiple fault - If the testing experiments does not detect the every single faults, then the circuit may contain one of the undetected faults at any time, and the occurrence of a second single faults between two testing experiments creates a multiple fault.

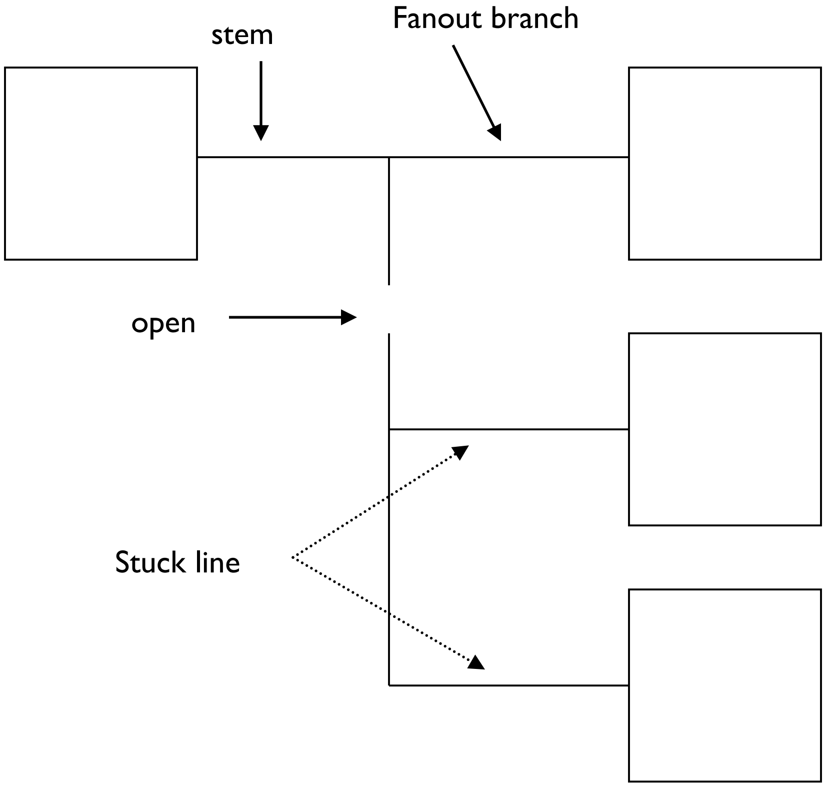
Stuck at Faults

- Faults affecting interconnections are **short & open**
- Short
  - By connecting points not intended to be connected
- Open
  - Breaking between connections
- A short between ground or supply and a signal line can make the signal remain at a fixed logic value v (v Є 0,1) and it is denoted by s-a-v
- A short between two signal lines usually creates a new logic function – **Bridging faults**

Stuck faults caused by open



Single stuck faults



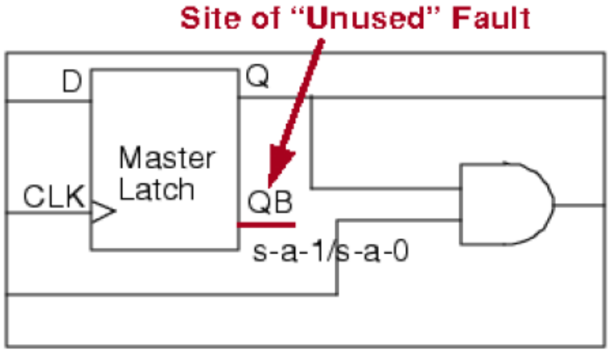
Multiple stuck faults

Fault Detection

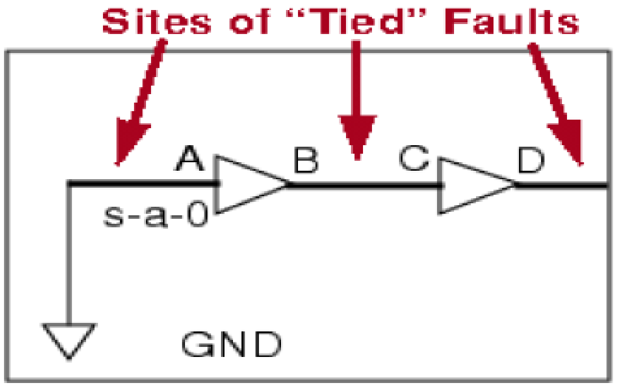
- Faults detection works by comparing the response of a known-good version of the circuit to that of the actual circuit, for a given stimulus set.
- A fault exists if there is any difference in the responses.
- One common fault detection approach is path sensitization.
- The path sensitization method, which is used by the tool to detect stuck-at faults, starts at the fault site and tries to construct a vector to propagate the fault effect to a primary output

Fault Classes

- **Untestable (UT)**
- Untestable (UT) faults are faults for which no pattern can exist to detect them. Untestable faults cannot cause functional failures, so the tools exclude them when calculating test coverage.
- The following are the Untestable Faults.
  - **Un-used (UU)**
    - The unused fault class includes all faults on circuitry unconnected to any circuit observation point and faults on floating primary outputs.

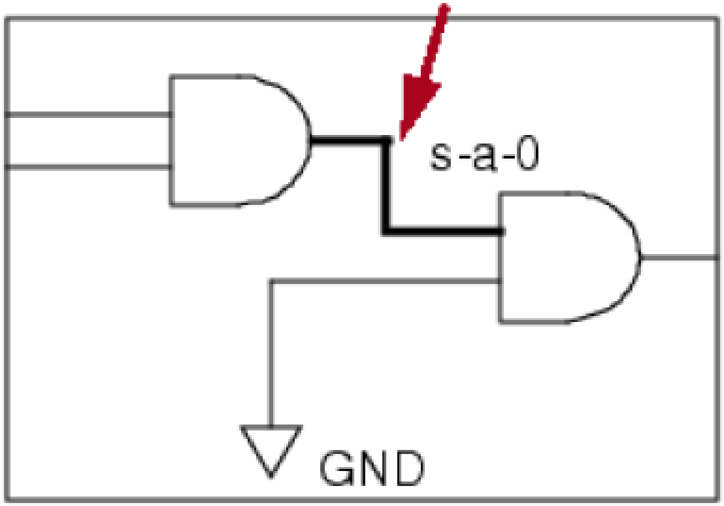


- **Tied (TI)**
- The tied fault class includes faults on gates where the point of the fault is tied to a value identical to the fault stuck value.
- In below fig, node A is connected to GND and there assumed to be a s-a-0 fault at A.



- **Blocked (BL)**
- The blocked fault class includes faults on circuitry for which tied logic blocks all paths to an observable point

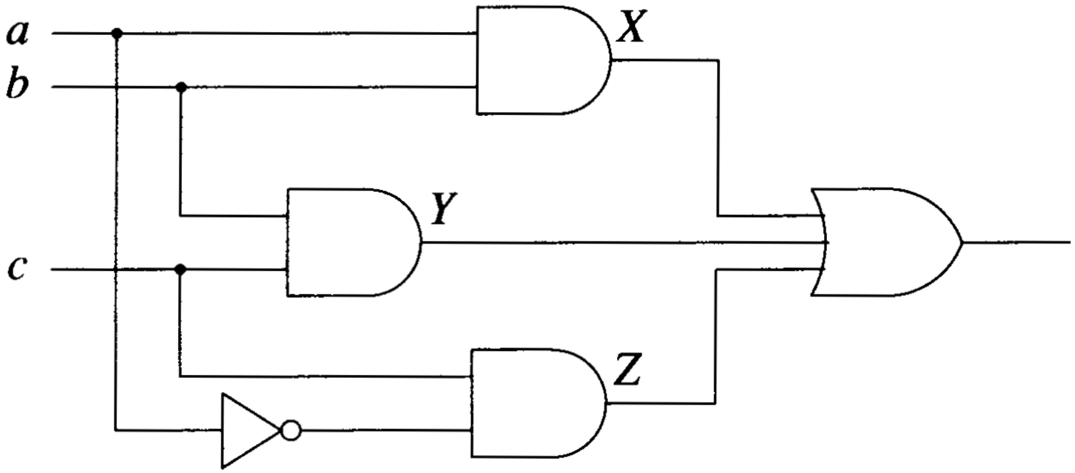
Site of "Blocked" Fault



Redundant Fault

- The redundant fault class includes faults that test generator considers undetectable.
- After the test pattern generator exhausts all patterns, it performs a special analysis to verify that the fault is undetectable under any conditions.

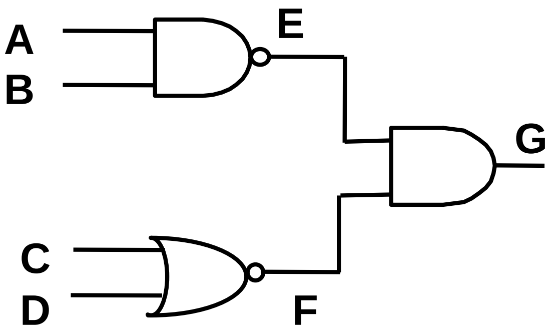
ab+bc+a'c



Undetectable if Y is s-a-0

Fault Modeling

- The effects of physical defects
- Most commonly used fault model: Single stuck-at fault



A s-a-1 B s-a-1 C s-a-1 D s-a-1
A s-a-0 B s-a-0 C s-a-0 D s-a-0

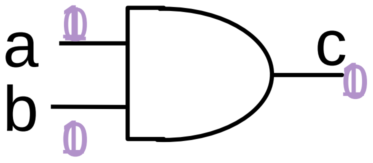
E s-a-1 F s-a-1 G s-a-1
E s-a-0 F s-a-0 G s-a-0

14 faults

Fault Coverage (FC)

$$FC = \frac{\text{\# faults detected}}{\text{\# faults in fault list}}$$

Example:



6 stuck-at faults
(a_0,a_1,b_0,b_1,c_0,c_1)

Test	faults detected	FC
{(0,0)}	c_1	16.67%
{(0,1)}	a_1,c_1	33.33%
{(1,1)}	a_0,b_0,c_0	50.00%
{(0,0),(1,1)}	a_0,b_0,c_0,c_1	66.67%
{(1,0),(0,1),(1,1)}	all	100.00%

Fault Modeling

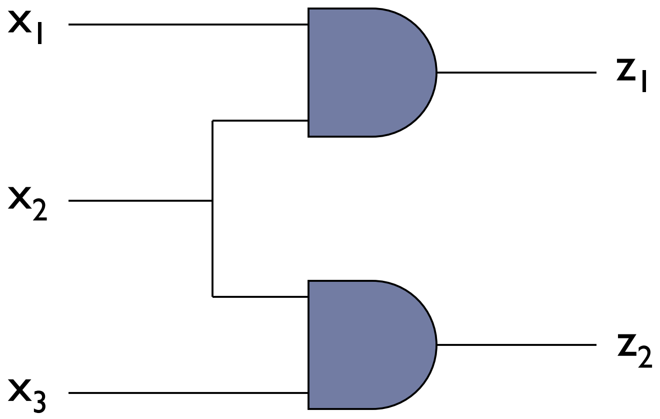
- Single stuck at faults
  - S @ 0
  - S @ 1
- Bridge fault
- Multiple stuck at faults

Single stuck at fault

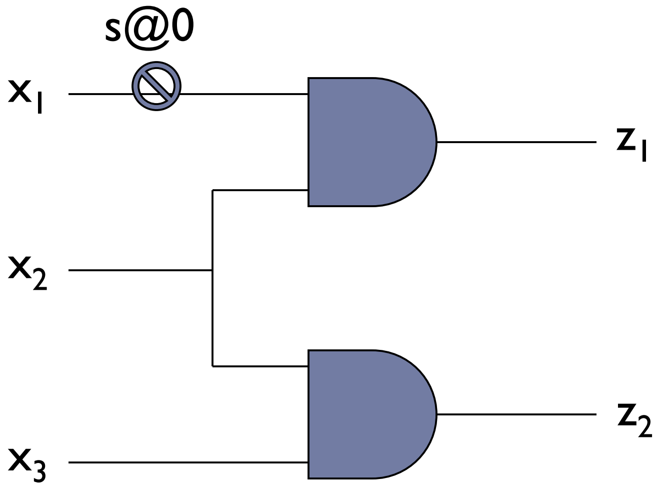
Fault free expression

Z1 = x1.x2

Z2 = x2.x3



Single stuck at fault - s@0



Fault free expression

$Z1 = x1.x2$

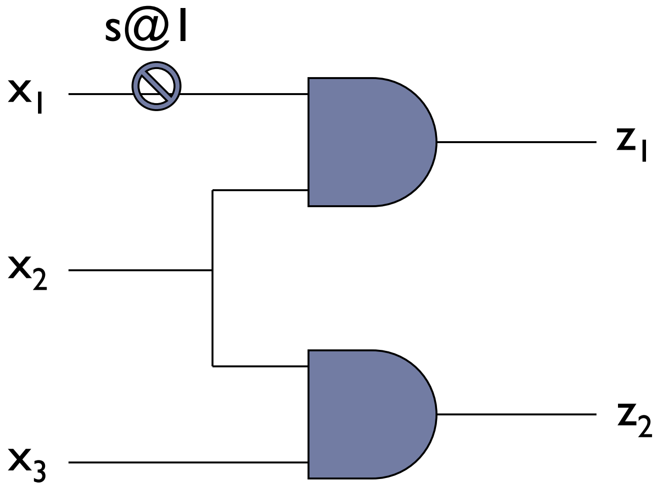
$Z2 = x2.x3$

Faulty expression

$Z1 = 0$

$Z2 = x2.x3$

Single stuck at fault - s@1



Fault free expression

$Z1 = x1.x2$

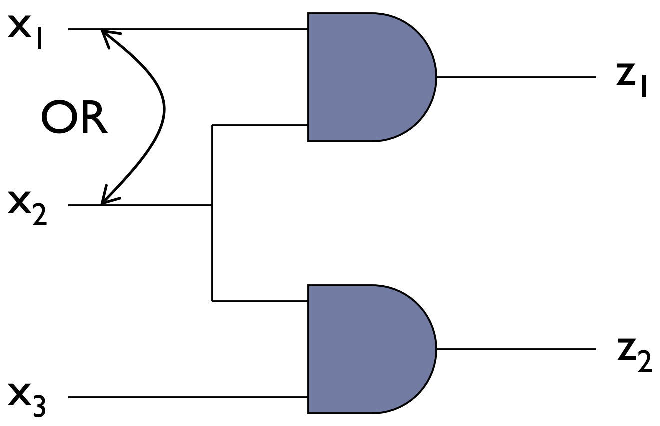
$Z2 = x2.x3$

Faulty expression

$Z1 = x2$

$Z2 = x2.x3$

Single stuck at fault - bridge fault



Fault free expression

Z1 = x1.x2

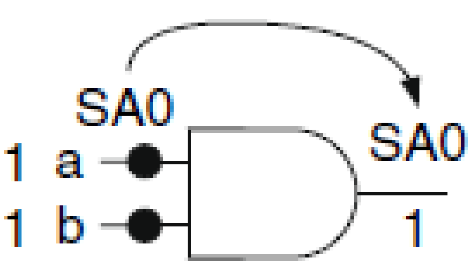
Z2 = x2.x3

Faulty expression

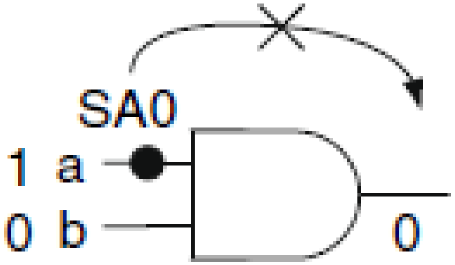
Z1 = (x1+x2).x2 = x2

Z2 = x2.x3

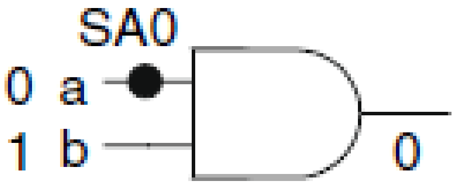
Fault Simulation Terminologies



a:SA0 propagates



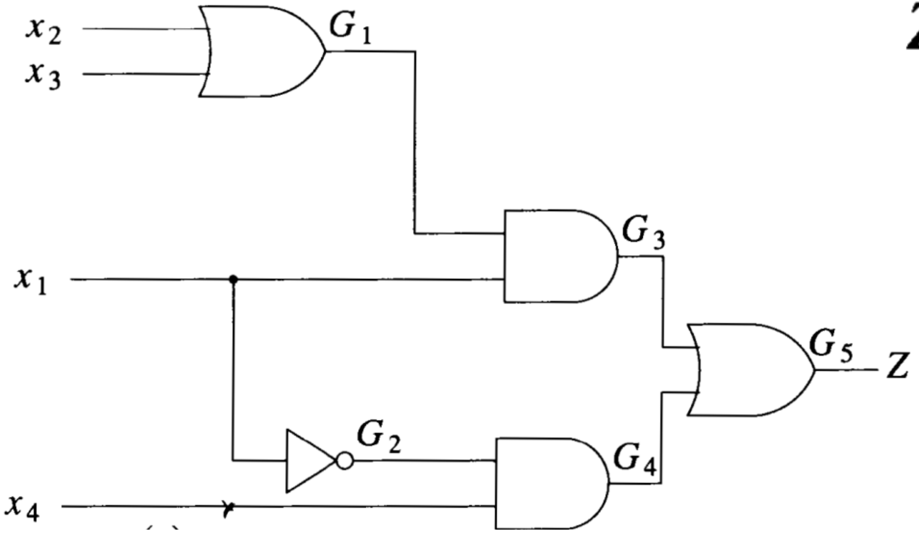
a:SA0 is blocked



a:SA0 is not activated

Fault Detection

- For a single output circuit, a test t that detects a fault f makes Z(t) = 0 and $Z_f(t) = 1$ or vice-versa.
- Thus set of all tests that detect f is given by the solutions of the equation
  - $Z(x)$ XOR $Z_f(x) = 1$



$$Z = (x_2+x_3)x_1 + \bar{x}_1 x_4$$

Let f be x_4 *s-a-0*.

$$Z_f = (x_2+x_3)x_1$$

$$\bar{x}_1 x_4 = 1.$$

Thus any test in which $x_1 = 0$ and $x_4 = 1$ is a test for f. any of the four tests (0001, 0011, 0101, 0111) that detect f.