ज्ञानेन प्रकाशते जगत्
INDUS UNIVERSITY

ELECTRONICS & COMMUNICATION ENGINEERING
IITE Ahmedabad

VLSI Design (EC0604)

Unit-4

B.Tech (Electronics and Communication)

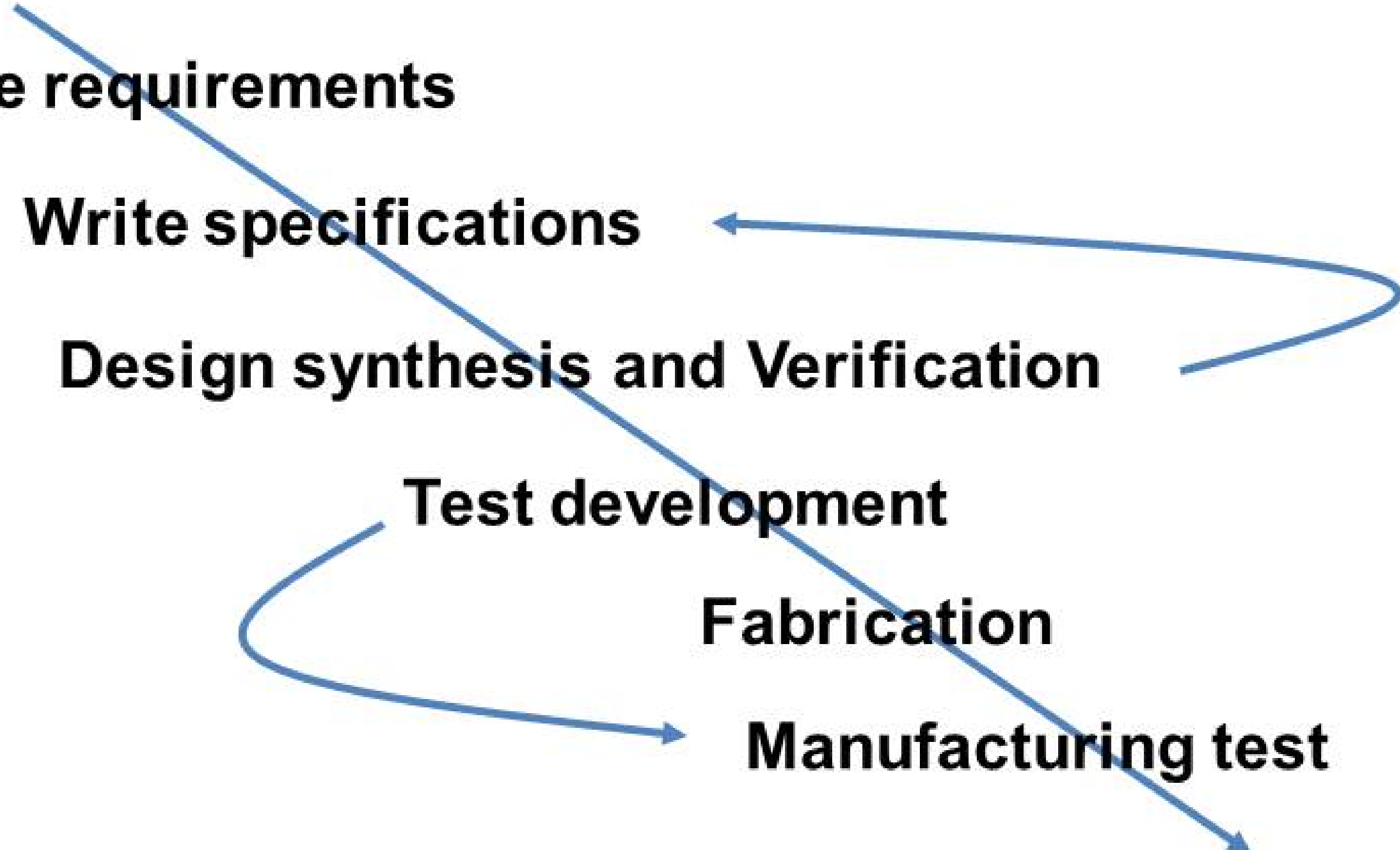
Semester-6

Ankur Changela

Introduction

- VLSI realization process
- Verification and test
- Ideal and real tests
- Costs of testing
- Roles of testing
- A modern VLSI device - system-on-a-chip
- Testing
  - Digital
  - Memory
  - Analog
  - RF
- Textbook
- Problem to solve

VLSI Realization Process



Definitions

- ***Design synthesis:*** **Given an I/O function, develop a procedure to manufacture a device using known materials and processes.**
- ***Verification:*** **Predictive analysis to ensure that the synthesized design, when manufactured, will perform the given I/O function.**
- ***Test:*** **A manufacturing step that ensures that the physical device, manufactured from the synthesized design, has no manufacturing defect.**

Verification vs. Test

Verification	*Test*
• Verifies correctness of design.	• Verifies correctness of manufactured hardware.
• Performed by simulation, hardware emulation, or formal methods.	• Two-part process: – 1. Test generation: software process executed once during design – 2. Test application: electrical tests applied to hardware
• Performed once prior to manufacturing.	• Test application performed on every manufactured device.
• Responsible for quality of design.	• Responsible for quality of devices.

Costs of Testing

- ***Design for testability*** **(DFT)**
  - **Chip area overhead and yield reduction**
  - **Performance overhead**
- **Software processes of test**
  - **Test generation and fault simulation**
  - **Test programming and debugging**
- **Manufacturing test**
  - ***Automatic test equipment*** **(ATE) capital cost**
  - **Test center operational cost**

Present and Future*

	1997 -2001	2003 - 2006
Feature size (micron)	**0.25 - 0.15**	**0.13 - 0.10**
Transistors/sq. cm	**4 - 10M**	**18 - 39M**
Pin count	100 – 900	160 - 1475
Clock rate (MHz)	**200 – 730**	**530 - 1100**
Power (Watts)	**1.2 – 61**	**2 - 96**

**** SIA Roadmap, IEEE Spectrum, July 1999***

Fault Modeling

- Why model faults?
- Some real defects in VLSI and PCB
- Common fault models
- Stuck-at faults
  - Single stuck-at faults
  - Fault equivalence
  - Fault dominance and checkpoint theorem
  - Classes of stuck-at faults and multiple faults
- Transistor faults
- Summary

Why Model Faults?

- I/O function tests inadequate for manufacturing (functionality versus component and interconnect testing)
- Real defects (often mechanical) too numerous and often not analyzable
- A fault model identifies targets for testing
- A fault model makes analysis possible
- Effectiveness measurable by experiments

Some Real Defects in Chips

- Processing defects
  - Missing contact windows
  - Parasitic transistors
  - Oxide breakdown
  - . . .
- Material defects
  - Bulk defects (cracks, crystal imperfections)
  - Surface impurities (ion migration)
  - . . .
- Time-dependent failures
  - Dielectric breakdown
  - Electromigration
  - . . .
- Packaging failures
  - Contact degradation
  - Seal leaks
  - . . .

Ref.: M. J. Howes and D. V. Morgan, *Reliability and Degradation - Semiconductor Devices and Circuits*, Wiley, 1981.

Observed PCB Defects

Defect classes	Occurrence frequency (%)
Shorts	51
Opens	1
Missing components	6
Wrong components	13
Reversed components	6
Bent leads	8
Analog specifications	5
Digital logic	5
Performance (timing)	5

Ref.: J. Bateson, *In-Circuit Testing*, Van Nostrand Reinhold, 1985.

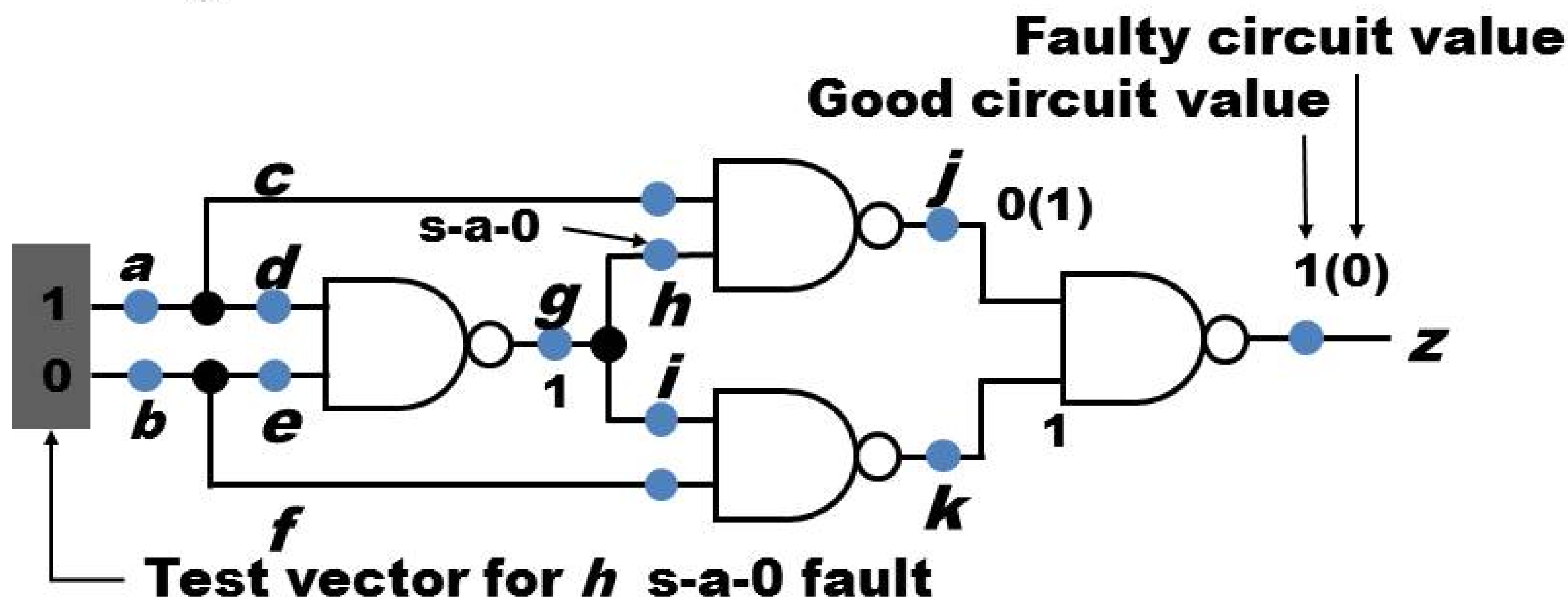
Common Fault Models

- **Single stuck-at faults**
- **Transistor open and short faults**
- **Memory faults**
- **PLA faults (stuck-at, cross-point, bridging)**
- **Functional faults (processors)**
- **Delay faults (transition, path)**
- **Analog faults**
- **For more details of fault models, see**

  M. L. Bushnell and V. D. Agrawal, *Essentials of Electronic Testing for Digital, Memory and Mixed-Signal VLSI Circuits*, Springer, 2000.

Single Stuck-at Fault

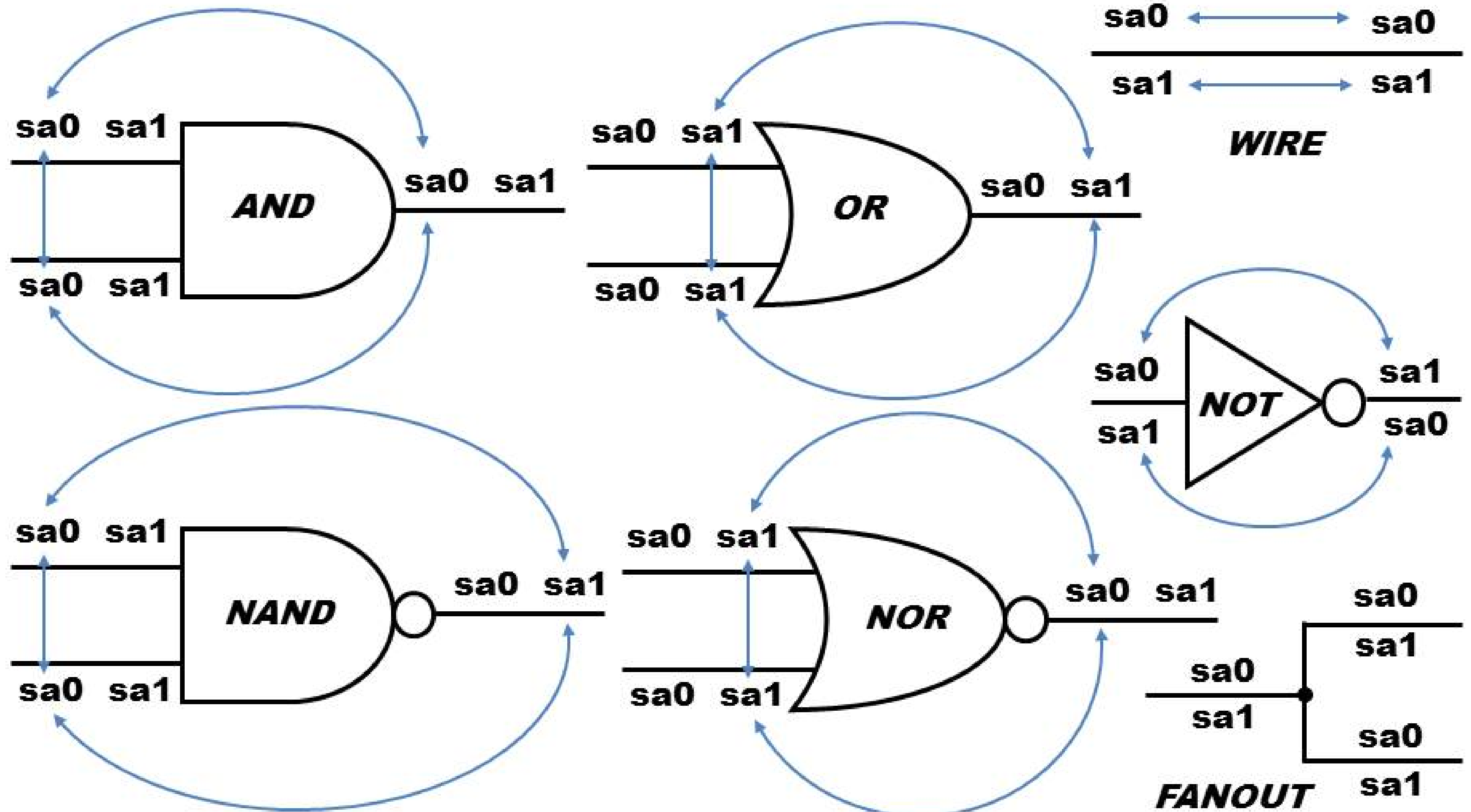
- **Three properties define a single stuck-at fault**
  - Only one line is faulty
  - The faulty line is permanently set to 0 or 1
  - The fault can be at an input or output of a gate
- **Example: XOR circuit has 12 fault sites ( ) and 24 single stuck-at faults**



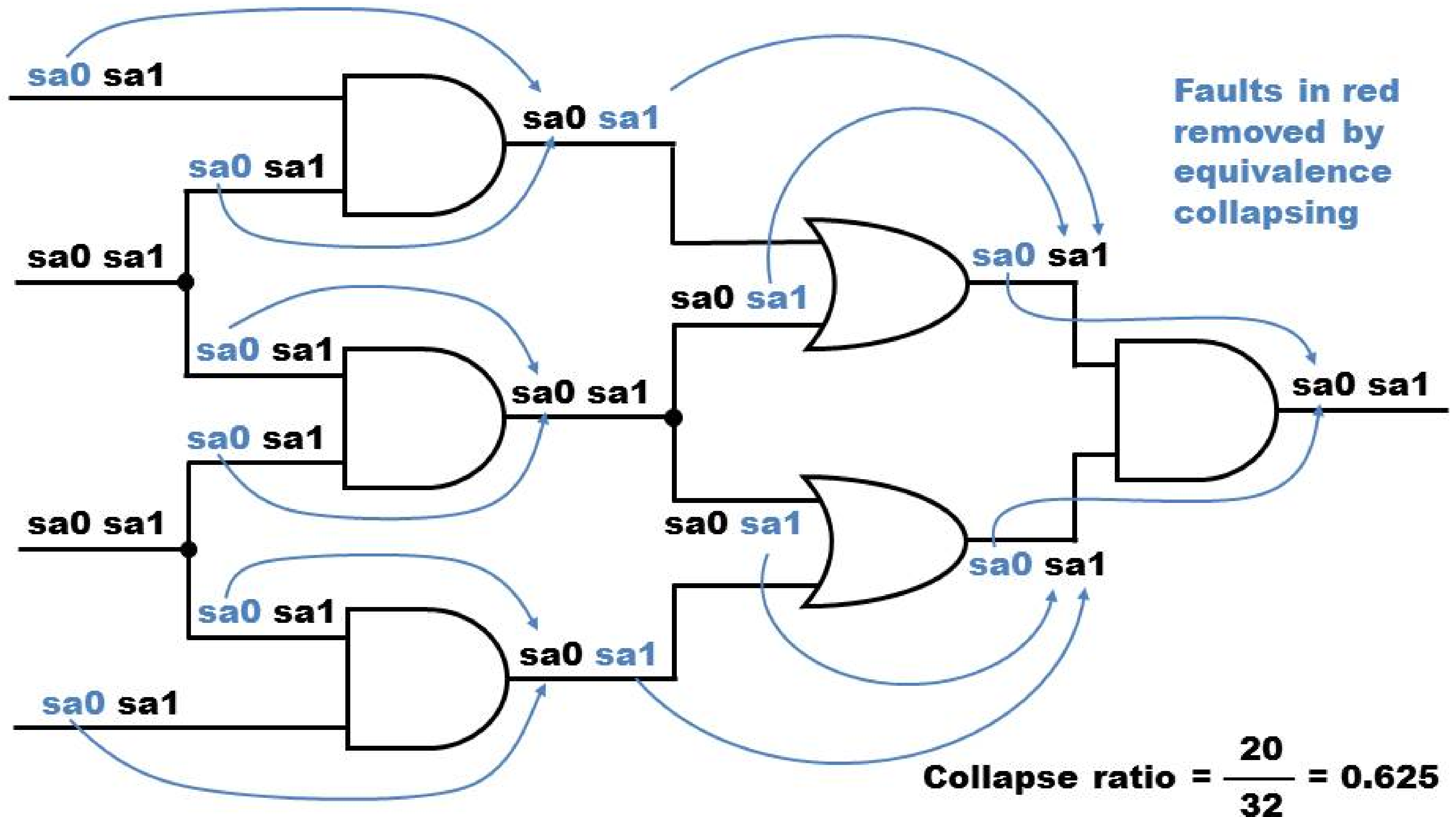
Fault Equivalence

- **Number of fault sites in a Boolean gate circuit is**
  = #PI + #gates + # (fanout branches)
- **Fault equivalence: Two faults f1 and f2 are equivalent if all tests that detect f1 also detect f2.**
- **If faults f1 and f2 are equivalent then the corresponding faulty functions are identical.**
- **Fault collapsing: All single faults of a logic circuit can be divided into disjoint equivalence subsets, where all faults in a subset are mutually equivalent. A collapsed fault set contains one fault from each equivalence subset.**

Structural Equivalence



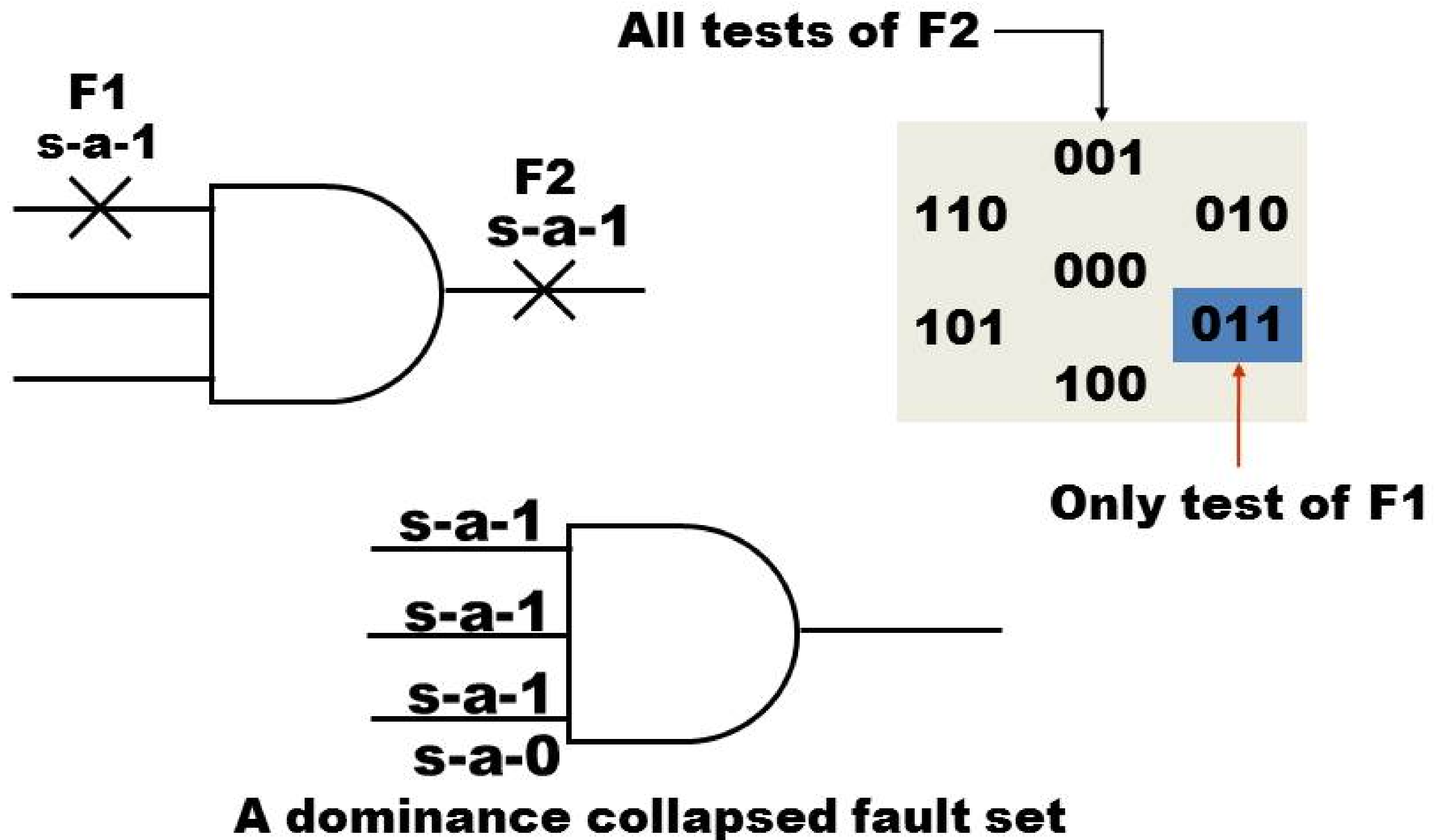
Equivalence Example



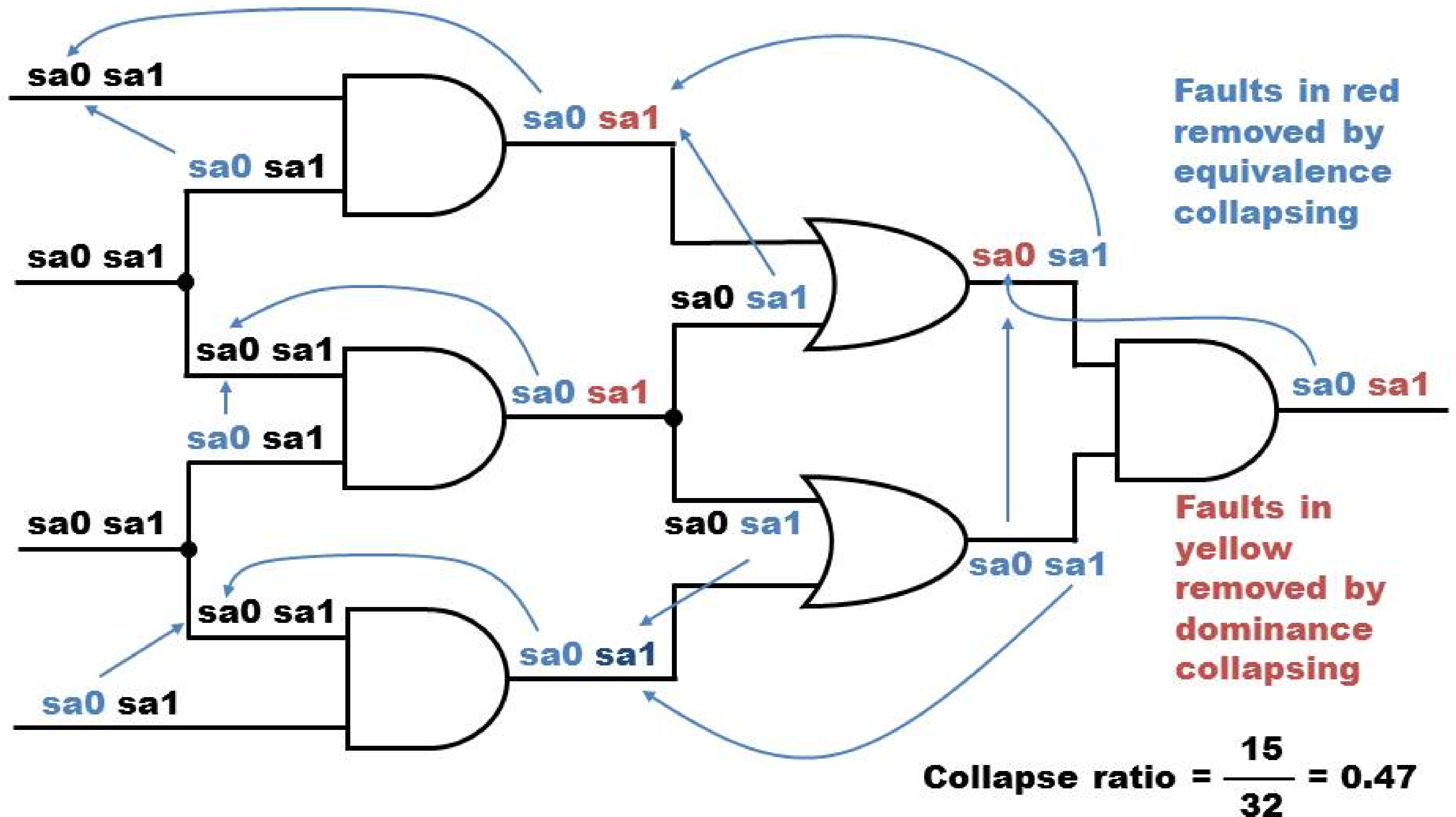
Fault Dominance

- **If all tests of some fault F1 detect another fault F2, then F2 is said to dominate F1.**
- **Dominance fault collapsing: If fault F2 dominates F1, then F2 is removed from the fault list.**
- **When dominance fault collapsing is used, it is sufficient to consider only the input faults of Boolean gates. See the next example.**
- **In a tree circuit (without fanouts) PI faults form a dominance collapsed fault set.**
- **If two faults dominate each other then they are equivalent.**

Dominance Example



Dominance Example



Lecture 4: Testability Analysis

- **Definition**
- **Controllability and observability**
- **SCOAP measures**
  - **Combinational circuits**
  - **Sequential circuits**
- **Summary**

What are Testability Measures?

- **Approximate measures of:**
  - **Difficulty of setting internal circuit lines to 0 or 1 from primary inputs.**
  - **Difficulty of observing internal circuit lines at primary outputs.**
- **Applications:**
  - **Analysis of difficulty of testing internal circuit parts – redesign or add special test hardware.**
  - **Guidance for algorithms computing test patterns – avoid using hard-to-control lines.**

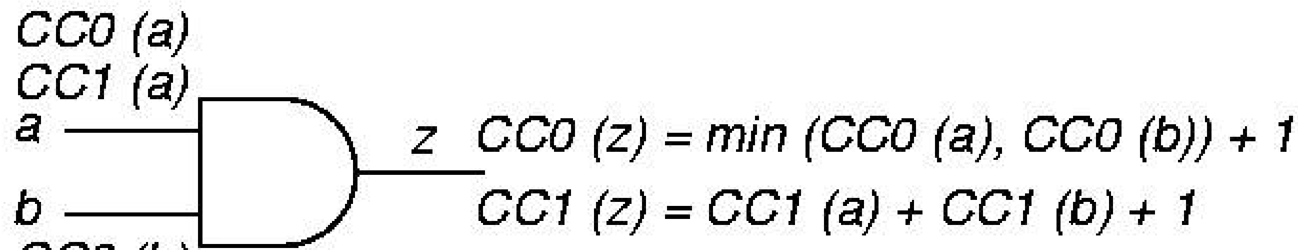
SCOAP Measures

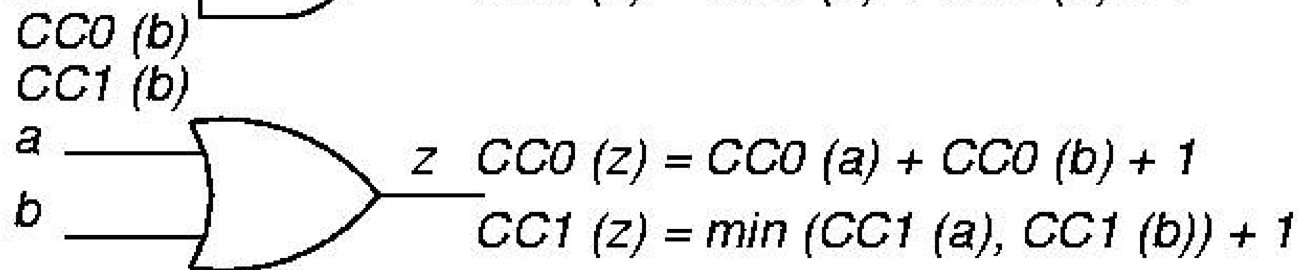
- **SCOAP – Sandia Controllability and Observability Analysis Program**
- **Combinational measures:**
  - ***CC0*** **– Difficulty of setting circuit line to logic 0**
  - ***CC1*** **– Difficulty of setting circuit line to logic 1**
  - ***CO*** **– Difficulty of observing a circuit line**
- **Sequential measures – analogous:**
  - ***SC0***
  - ***SC1***
  - ***SO***
- **Ref.: L. H. Goldstein, "Controllability/Observability Analysis of Digital Circuits,"** ***IEEE Trans. CAS,*** **vol. CAS-26, no. 9. pp. 685 – 693, Sep. 1979.**

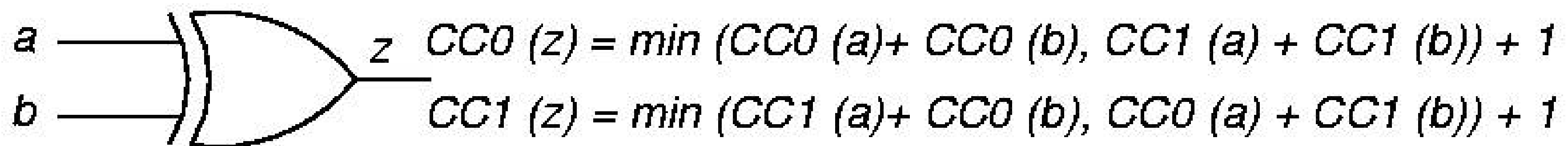
Range of SCOAP Measures

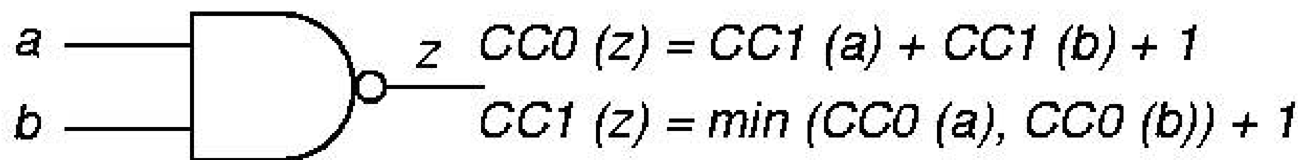
- **Controllabilities – 1 (easiest) to infinity (hardest)**
- **Observabilities – 0 (easiest) to infinity (hardest)**
- **Combinational measures:**
  - **Roughly proportional to number of circuit lines that must be set to control or observe given line.**
- **Sequential measures:**
  - **Roughly proportional to number of times flip-flops must be clocked to control or observe given line.**

Combinational Controllability

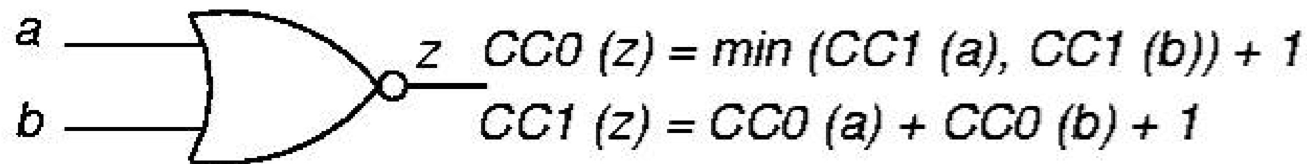


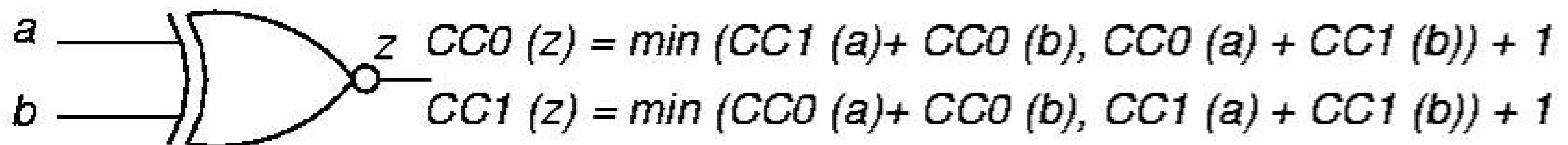


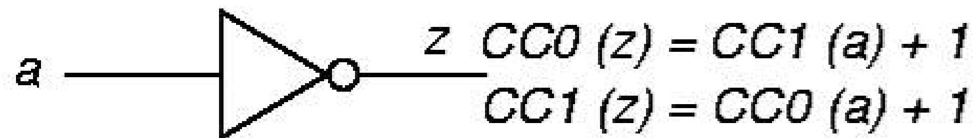




Controllability Formulas (Continued)







Combinational Observability

To observe a gate input: Observe output and make other input values non-controlling.

$CO(a) = CO(z) + CC1(b) + 1$

$CO(b) = CO(z) + CC1(a) + 1$

$CO(a) = CO(z) + CC0(b) + 1$

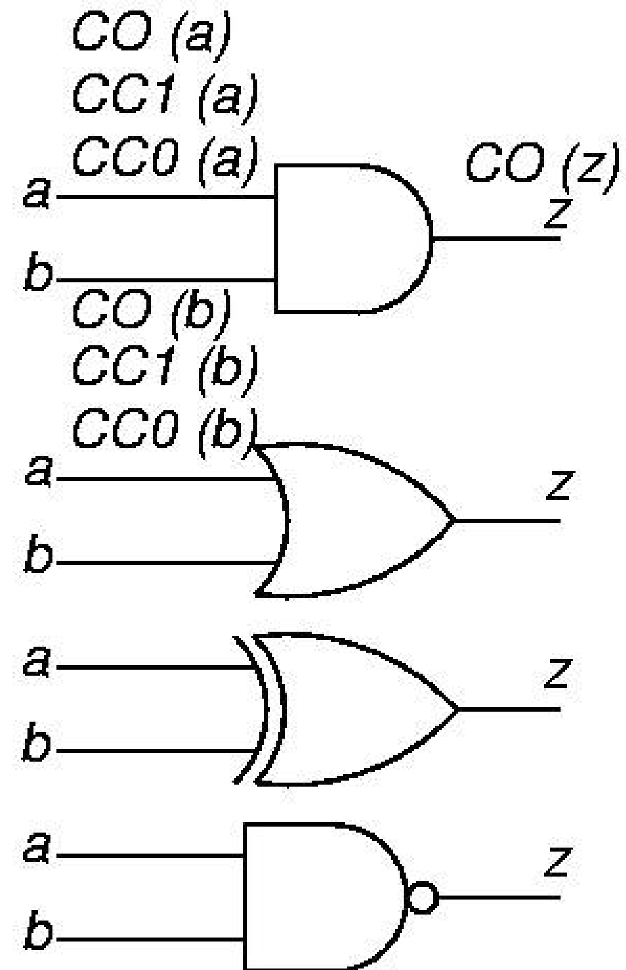
$CO(b) = CO(z) + CC0(a) + 1$

$CO(a) = CO(z) + \min(CC0(b), CC1(b)) + 1$

$CO(b) = CO(z) + \min(CC0(a), CC1(a)) + 1$

$CO(a) = CO(z) + CC1(b) + 1$

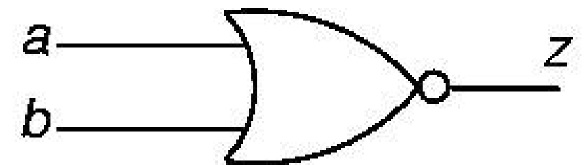
$CO(b) = CO(z) + CC1(a) + 1$



Observability Formulas (Continued)

$CO(a) = CO(z) + CC0(b) + 1$

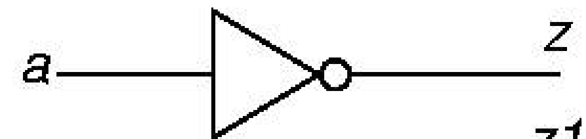
$CO(b) = CO(z) + CC0(a) + 1$



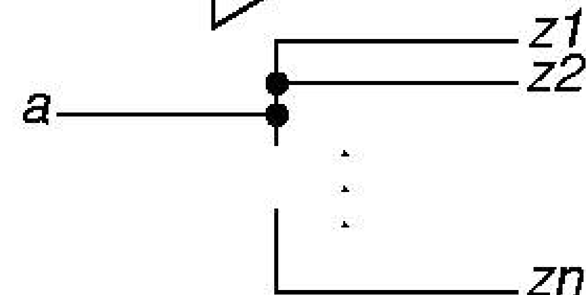
$CO(a) = CO(z) + \min(CC0(b), CC1(b)) + 1$

$CO(b) = CO(z) + \min(CC0(a), CC1(a)) + 1$

$CO(a) = CO(z) + 1$



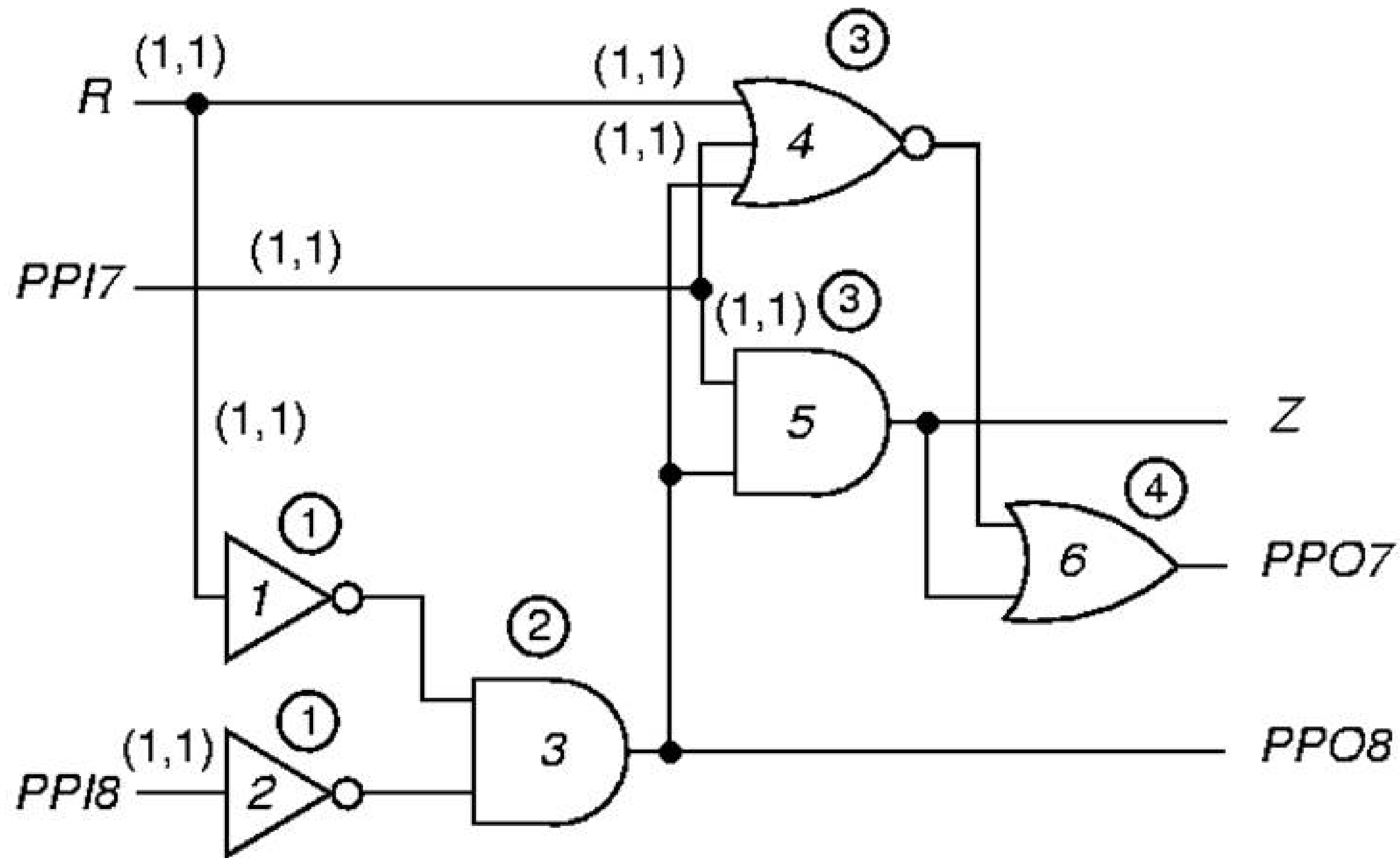
$CO(a) = \min(CO(z1), CO(z2), ..., CO(zn))$



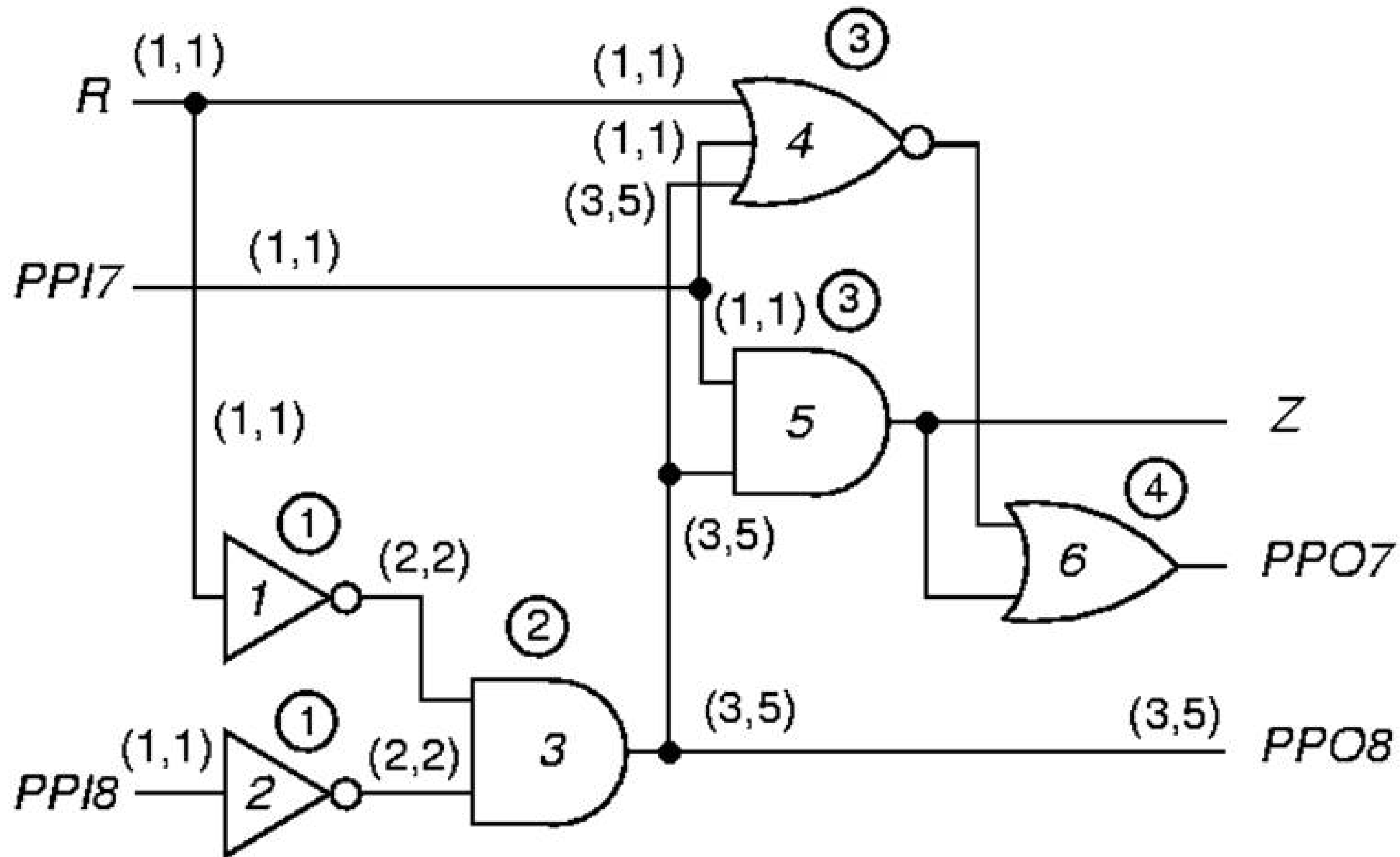
Fanout stem: Observe through branch with best observability.

Comb. Controllability

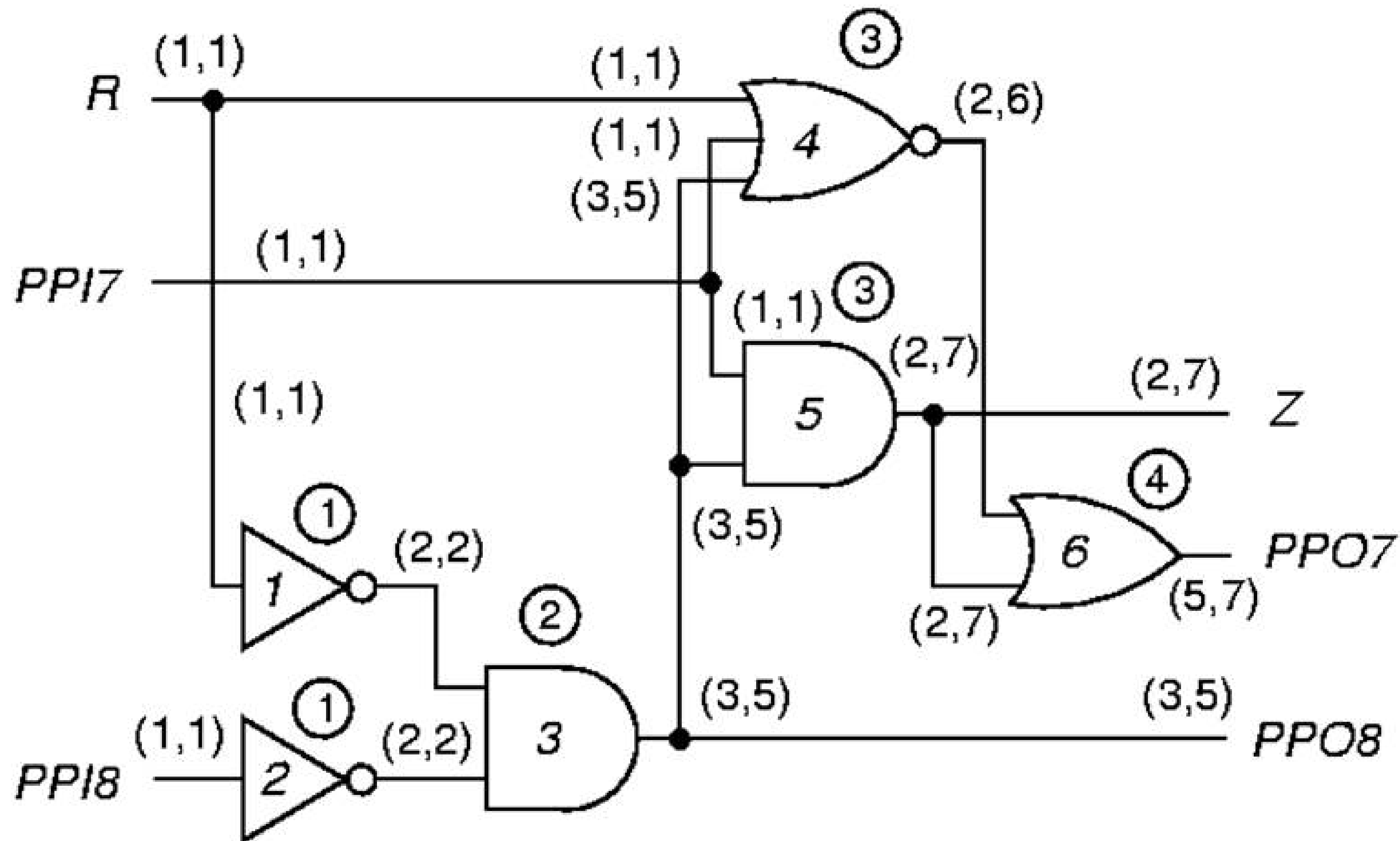
Circled numbers give level number. (CC0, CC1)



Controllability Through Level 2

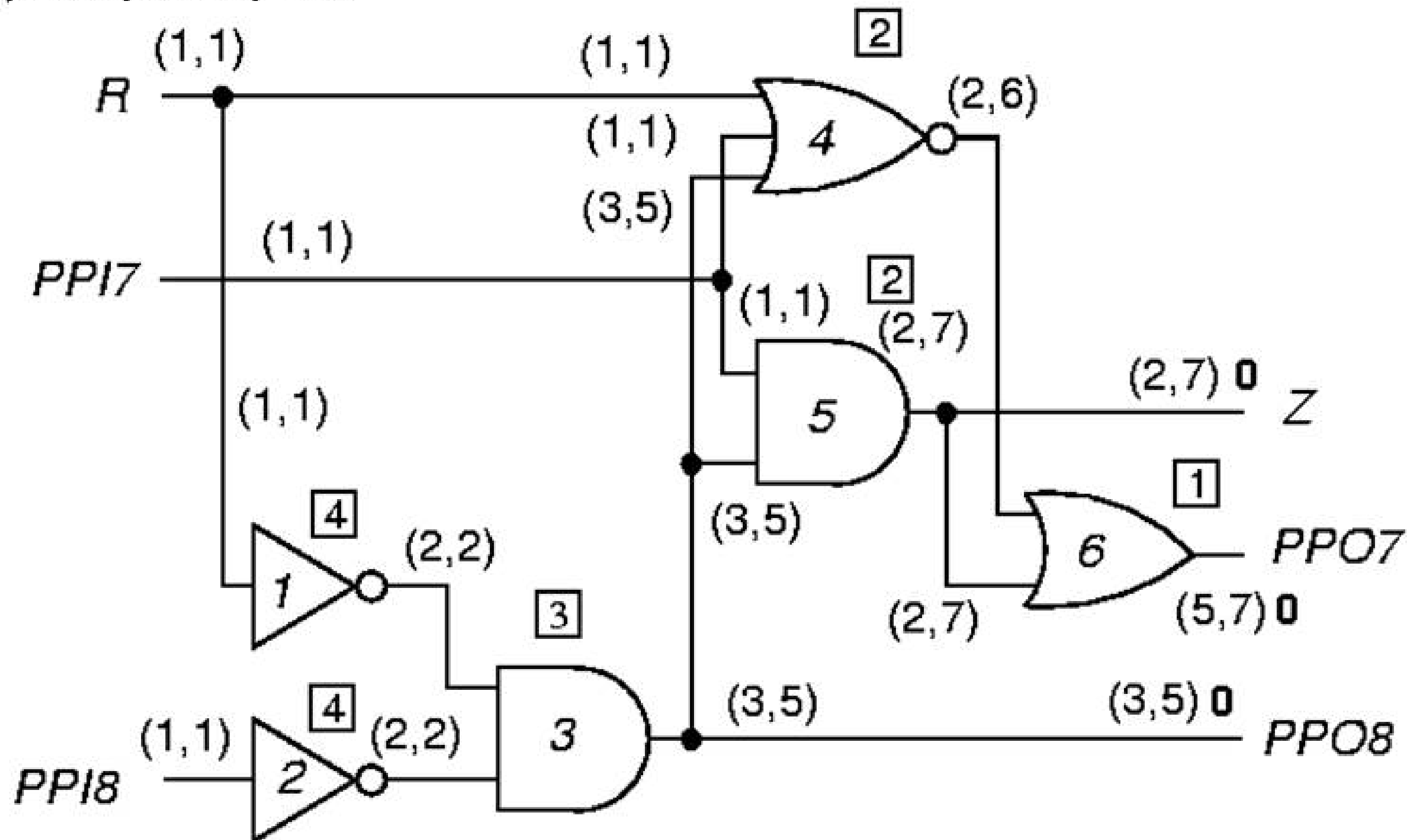


Final Combinational Controllability

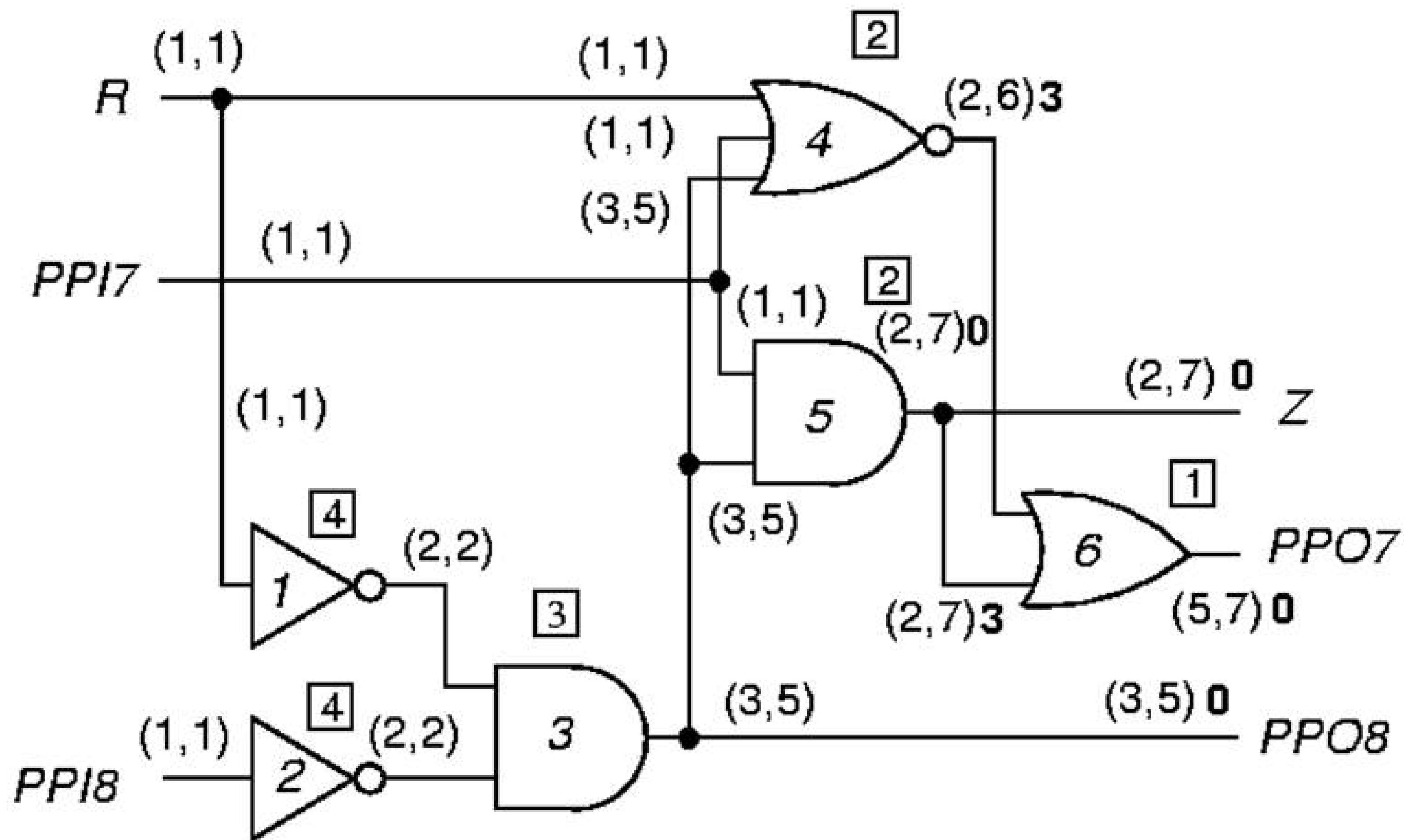


Combinational Observability for Level 1

Number in square box is level from *primary outputs* (POs).
(CC0, CC1) CO



Combinational Observabilities for Level 2



Final Combinational Observabilities