DIGITAL SYSTEM TESTING
COE -545

Lecture – 08

Testability Measures

Objective

- Regrettably, Decisions in the D-ALG are *Arbitrary*
- **PODEM** Uses the Testability of Nodes to help Choose among alternative Choices.
- *Derive* an *Approximate Measure* Of How Easy/Hard it is to Test a Given CUT
- **Usage**:
  - Analyze the Difficulty of Testing a given CUT → *Redesign or Add Special Test Hardware*
  - Guidance for Algorithms Computing Test Patterns → *Make Better Informed Decisions in the D-Drive and Line-Justification Steps*
  - Estimation of Fault Coverage
  - Estimation of Test Vector Length

COE – KFUPM *Dr. Alaaeldin Amin (COE 545)* *Slide Number 2*

Testability Measures

- Generally 2 Measures are Used:
  1. Controllability (CY), and
  2. Observability (OY),
- No Agreed-Upon Standard for measuring CY or OY {SCOAP, CAMELOT, VICTOR, etc.}

Controllability (CY):

Defined for Both Logic 1 & 0; as a measure for the Ability to Control the State of a Given Line (node) from PIs → Thus, Every Line Has a 1-Controllability and a 0-Controllability

Observability (OY):

A measure of the Ease/Difficulty with which the State of a given Line can be Observed at one or more PO

- CY & OY May be Normalized Between 0 & 1.

COE – KFUPM *Dr. Alaaeldin Amin (COE 545)* *Slide Number 3*

Testability Analysis

- Involves Circuit *Topological analysis*, but no test vectors and *no search* algorithm
  - *Static analysis*
- Linear computational complexity
  - Otherwise, is pointless – might as well use automatic test-pattern generation and calculate:
    - Exact fault coverage
    - Exact test vectors

COE – KFUPM *Dr. Alaaeldin Amin (COE 545)* *Slide Number 4*

SCOAP – Sandia Controllability and Observability Analysis Program

- Computes Combinational & Sequential CY & OY
- 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*

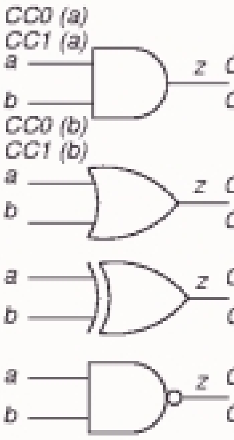
COE – KFUPM *Dr. Alaaeldin Amin (COE 545)* *Slide Number 5*

Range of SCOAP Measures

- Each Line is Assigned Controllability & Observability *Numeric* Values
- **Controllability – 1** (easiest to control) to **infinity** (hardest)
- **Observability – 0** (easiest to observe) to **infinity** (hardest)
- Combinational measures:
  - Roughly Proportional to # Circuit Lines that must be Set to Control or Observe the Given Line
  - Line Controllabilities are Computed **Starting** at the **PIs** moving *Forward* to **POs**
  - Line Observabilities are Computed **Starting** at the **POs** moving *Backwards* to **PIs**
- Sequential measures:
  - Roughly proportional to # times a flip-flop must be clocked to control or observe given line

COE – KFUPM *Dr. Alaaeldin Amin (COE 545)* *Slide Number 6*

Controllability Examples



Only One I/P Needs to be Controlled

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

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

All I/Ps Must be Controlled

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

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

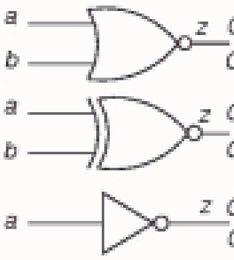
$CC0(z) = \min(CC0(a) + CC0(b), CC1(a) + CC1(b)) + 1$

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

$CC0(z) = CC1(a) + CC1(b) + 1$

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

Controllability Examples (Contd)



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

$CC1(z) = CC0(a) + CC0(b) + 1$

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

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

$CC0(z) = CC1(a) + 1$

$CC1(z) = CC0(a) + 1$

Observability Examples

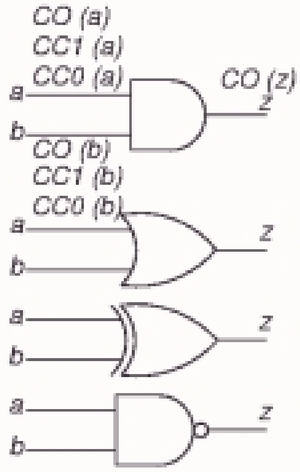
To observe a gate input: *Observe output and make other input values non-controlling*

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

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

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

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



COE – KFUPM *Dr. Alaaeldin Amin (COE 545)* *Slide Number 9*

More Observability Examples

To observe a *fanout stems*:
Observe it through branch with best observability

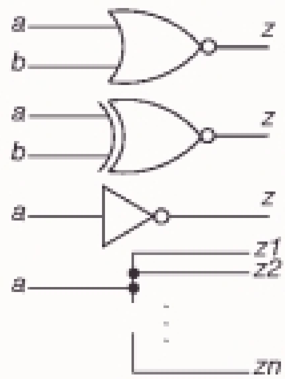
$CO(a) = CO(z) + CC0(b) + 1$
$CO(b) = CO(z) + CC0(a) + 1$

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

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

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

Min→ Independent Branches
Max→ Dependent Branches



COE – KFUPM *Dr. Alaaeldin Amin (COE 545)* *Slide Number 10*

Goldstein's SCOAP Measures

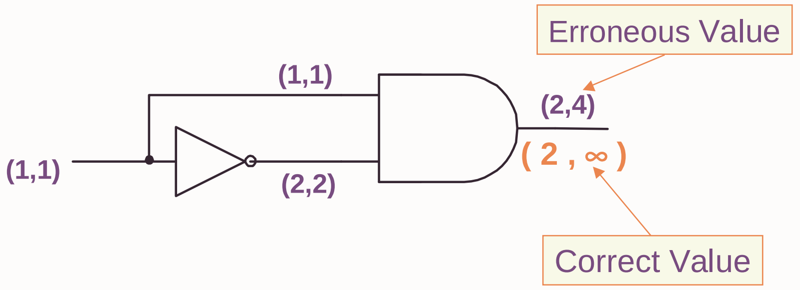
- **AND** gate **0-Controllability**:
  Output_Controllability = Min (Input_Controllability) **+ 1**
- **AND** gate **1-Controllability**:
  Output_Controllability = **Σ** (Input_Controllability) **+ 1**

- **XOR** gate O/P **Controllability**
  Output_Controllability = **min** (controllabilities of each input set) **+ 1**

- **Fanout Stem Observability**:
  Σ or **min** (some or all fanout branch observabilities)

Error Due to Stems & Reconverging Fanouts

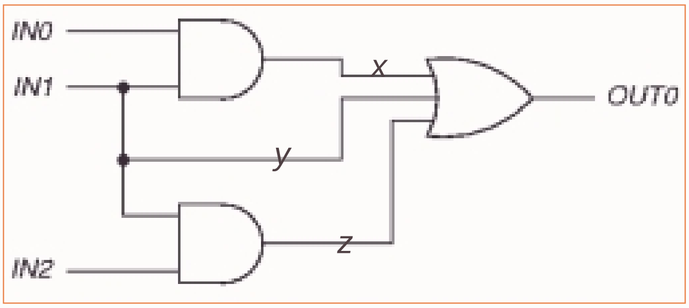
SCOAP measures Incorrectly Computes Controllabilities Assuming that Fanout Branches to Be Independent → Untrue for Reconvergent Branches



Error Due to Stems & Reconverging Fanouts

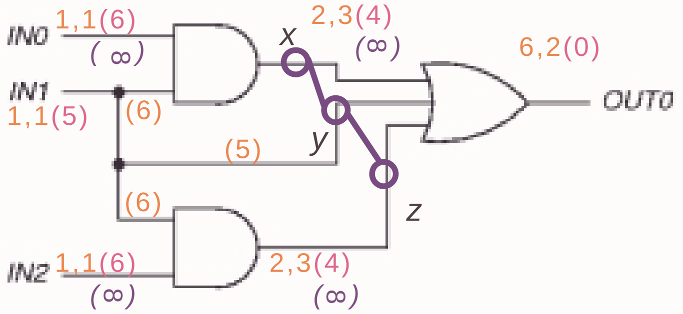
SCOAP measures Incorrectly assume that controlling or observing *x, y, z* are *independent* events

- ***CC0 (x), CC0 (y), CC0 (z)* correlate**
- ***CC1 (x), CC1 (y), CC1 (z)* correlate**
- ***CO (x), CO (y), CO (z)* correlate**



Correlation Error Example

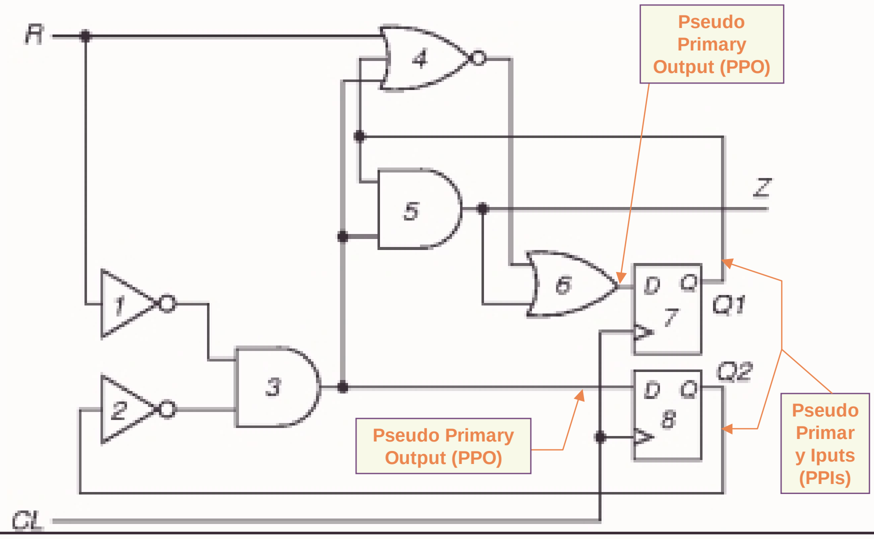
- **Exact computation of measures is NP-Complete and impractical**
- **SCOAP measures are in red or bold CC0,CC1 (CO) -- Italicized (Blue) measures show correct values**



Sequential Example

Assumption: Fully Controllable (Scannable) FFs→

- FF Inputs → Pseudo POs of Combinational Logic
- FF Outputs → Pseudo PIs for Combinational Logic



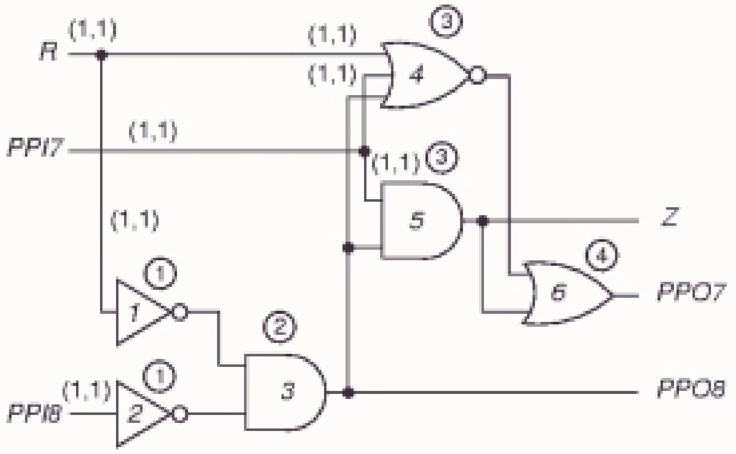
Levelization Algorithm 6.1

- **Label** each gate with max # of logic levels from primary inputs or with max # of logic levels from primary output
- **Assign** level # 0 to all *primary inputs* (PIs)
- **For each PI fanout**:
  - Label that line with the PI level number, &
  - Queue logic gate driven by that fanout
- **While queue is not empty**:
  - Dequeue the next logic gate
  - **IF** all gate inputs have level #'s,
    - label the gate with the highest of them + 1;
    - Label O/P Line & its Fanout Lines with the same level number, &
    - Queue logic gates driven by the fanout lines
  - **Else**, requeue the gate

ber 16

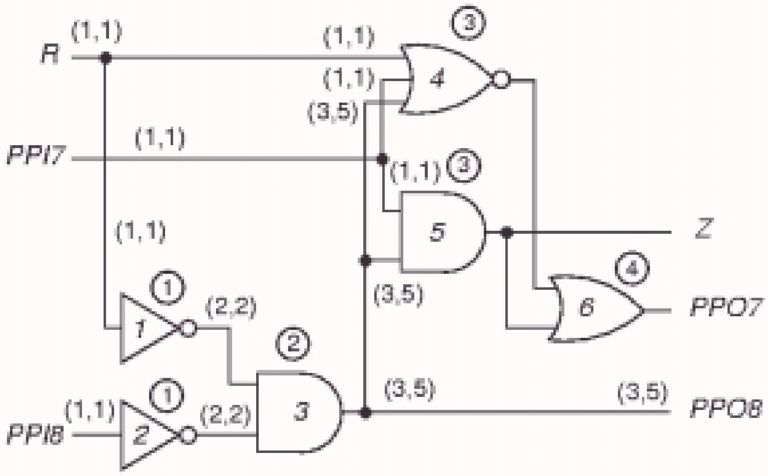
Controllability Through Level 0

Circled numbers → level number.
Parenthesized Numbers →(CC0, CC1)



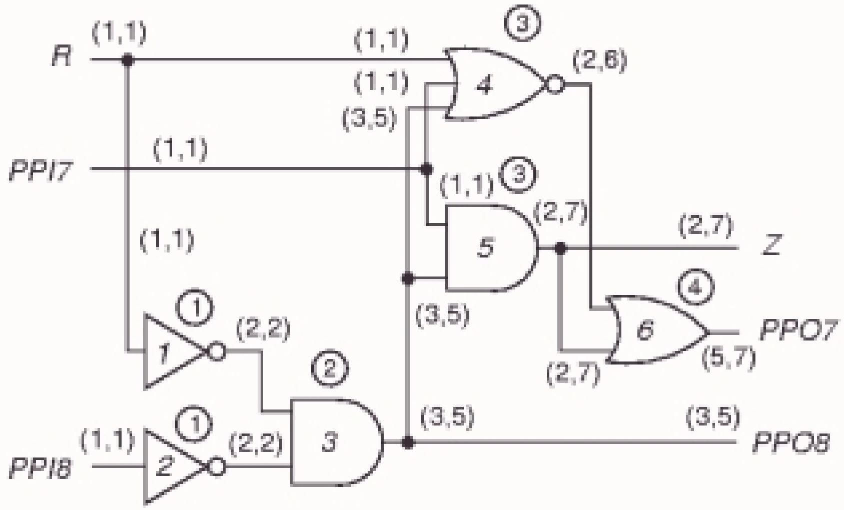
COE – KFUPM   Dr. Alaaeldin Amin (COE 545)   Slide Number 17

Controllability Through Level 2



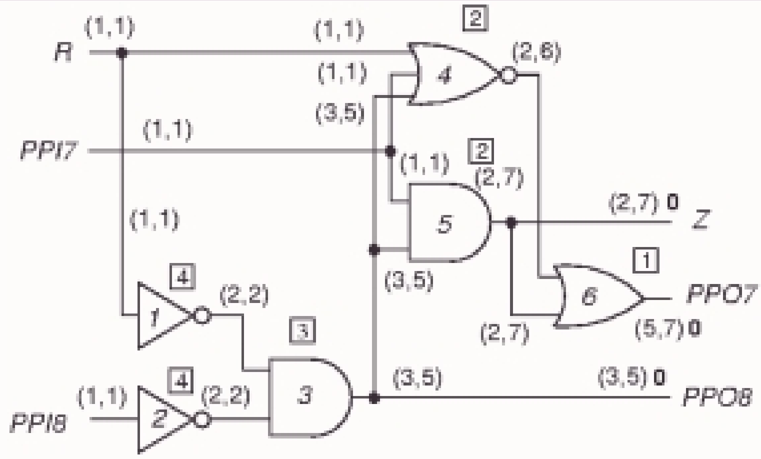
COE – KFUPM   Dr. Alaaeldin Amin (COE 545)   Slide Number 18

Final Combinational Controllability

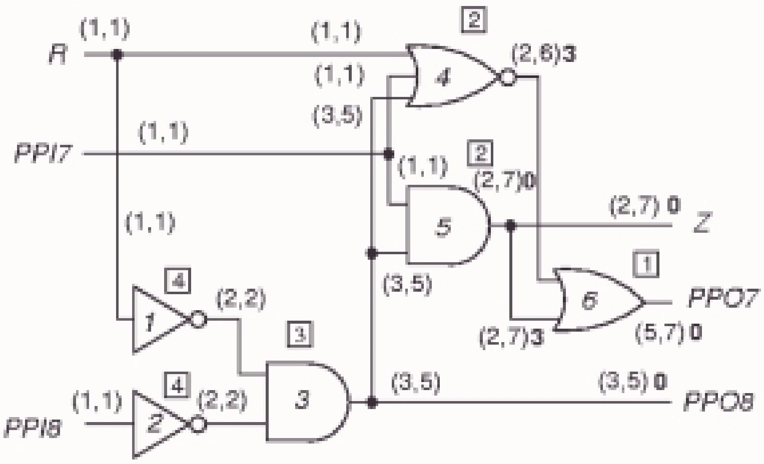


Combinational Observability for Level 1

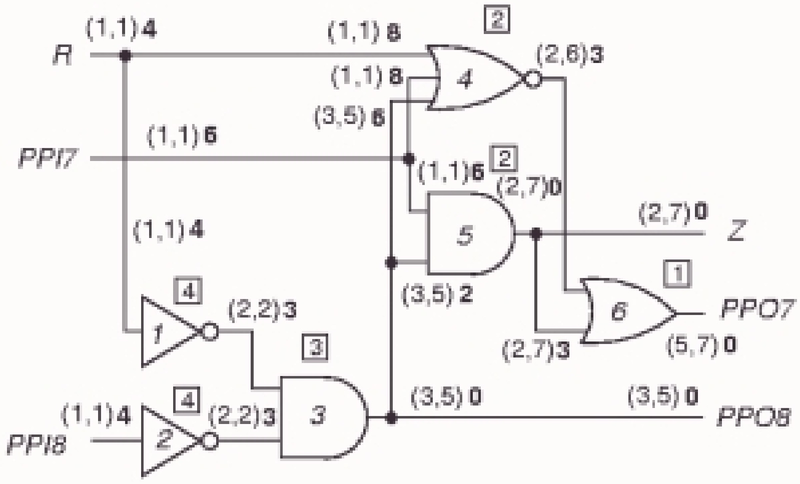
Number in square box →level from *primary outputs* (POs).
(CC0, CC1) CO



Combinational Observabilities for Level 2



Final Combinational Observabilities



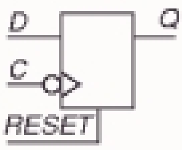
Sequential Measure Differences

- **Combinational**
  - Increment *CC0*, *CC1*, *CO* when passing through a Gate, either forwards (Controllability) or backwards(Observability)
  - CCi, CO *are not incremented when passing through* *FFs*, (CC0(Q), CC1(Q), CO(D)) either forwards or backwards
- **Sequential**
  - Increment *SCi*, *SO* only when Passing through a FF, either forwards or backwards, to *Q*, $\overline{Q}$, *D*, *C*, *SET*, or *RESET*
  - *SCi & SO are not incremented when passing through* *Combinational Gates*, either forwards or backwards.
- **Both**
  - Must iterate on feedback loops until controllabilities stabilize

COE – KFUPM   Dr. Alaaeldin Amin (COE 545)   Slide Number 23

D Flip-Flop Equations

- *Assume a* ***Synchronous*** *RESET line*.
- ***CC1* (*Q*) = *CC1* (*D*) + *CC1* (*C*) + *CC0* (*C*) + *CC0*(*RESET*)**
- ***SC1*(*Q*) = *SC1* (*D*) + *SC1* (*C*) + *SC0* (*C*) + *SC0*(*RESET*) +1**
- ***CC0* (*Q*) = min {*CC1* (*RESET*) + *CC1* (*C*) + *CC0* (*C*) , *CC0* (*D*) + *CC1* (*C*) + *CC0* (*C*)}**
- ***SC0* (*Q*) = min {*SC1* (*RESET*) + *SC1* (*C*) + *SC0* (*C*) , *SC0* (*D*) + *SC1* (*C*) + *SC0* (*C*)} + 1**
- ***CO* (*D*) = *CO* (*Q*) + *CC1* (*C*) + *CC0* (*C*) + *CC0*(*RESET*)**
- ***SO* (*D*) = *SO* (*Q*) + *SC1* (*C*) + *SC0* (*C*) + *SC0*(*RESET*) +1**



COE – KFUPM   Dr. Alaaeldin Amin (COE 545)   Number 24

D Flip-Flop Clock and Reset

- ***CO* (*RESET*) = *CO* (*Q*) + *CC1* (*Q*) + *CC1* (*RESET*) + *CC1* (*C*) + *CC0* (*C*)**
- ***SO* (*RESET*) = *SO* (*Q*) + *SC1* (*Q*) + *SC1* (*RESET*) + *SC1* (*C*) + *SC0* (*C*) + 1**
- **Three ways to observe the clock line:**
  1. **Set *Q* to 1 and clock in a 0 from *D***
  2. **Set the flip-flop and then reset it**
  3. **Reset the flip-flop and clock in a 1 from *D***
- ***CO* (*C*) = min { *CO* (*Q*) + *CC1* (*Q*) + *CC0* (*D*) + *CC1* (*C*) + *CC0* (*C*) ,**
  ***CO* (*Q*) + *CC1* (*Q*) + *CC1* (*RESET*) + *CC1* (*C*) + *CC0* (*C*) ,**
  ***CO* (*Q*) + *CC0* (*Q*) + *CC0* (*RESET*) + *CC1* (*D*) + *CC1* (*C*) + *CC0* (*C*) }**
- ***SO* (*C*) = Similar to *CO* (*C*) + 1**

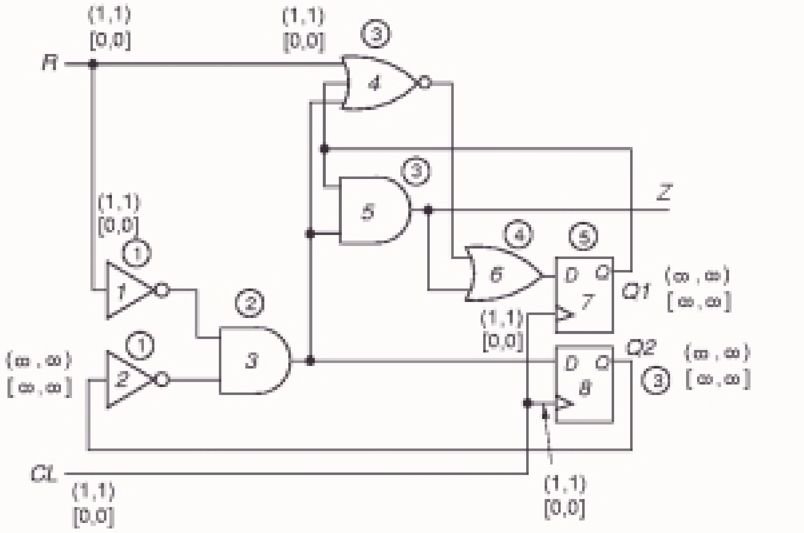
COE – KFUPM *Dr. Alaaeldin Amin (COE 545)* *Slide Number 25*

Algorithm: Testability Computation

1. For all **PIs**, *CC0* = *CC1* = 1 and *SC0* = *SC1* = 0
2. Initialize all other nodes, *CC0* = *CC1* = *SC0* = *SC1* = ∞
3. Go from **PIs** to **POs**, using *CC* and *SC* equations to get controllabilities -- ***Iterate*** on loops *until SC stabilizes* → Convergence Guaranteed {Monotonically Decreasing Values, (initially ∞)} → 2-3 Iterations are Typical.
4. For all **POs**, set *CO* = *SO* = **0** , For **Other Nodes** CO=SO=∞
5. Work from **POs** to **PIs**, Use *CO*, *SO*, and *Precomputed controllabilities* to map O/P Node Observabilities of Gates & FFS into I/P observabilities
6. Fanout stem (*CO*, SO) = **min {** branch (CO, SO)**}**
7. If a *CC* or *SC* (*CO* or *SO*) is ∞ , that node is uncontrollable **or** unobservable.

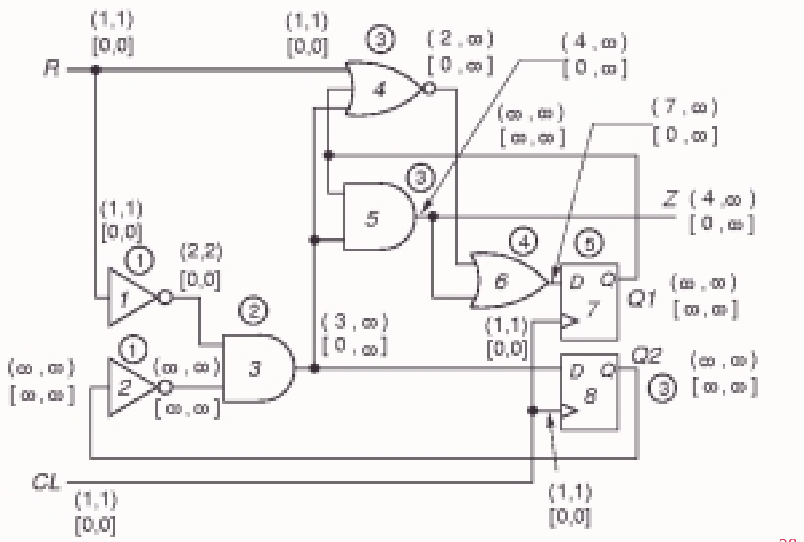
COE – KFUPM *Dr. Alaaeldin Amin (COE 545)* *Slide Number 26*

Sequential Example Initialization



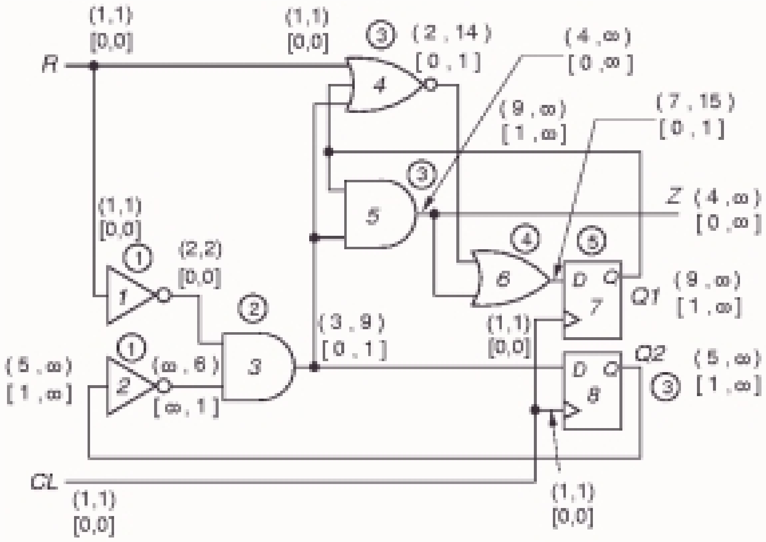
COE – KFUPM Dr. Alaaeldin Amin (COE 545) Slide Number 27

After 1 Iteration



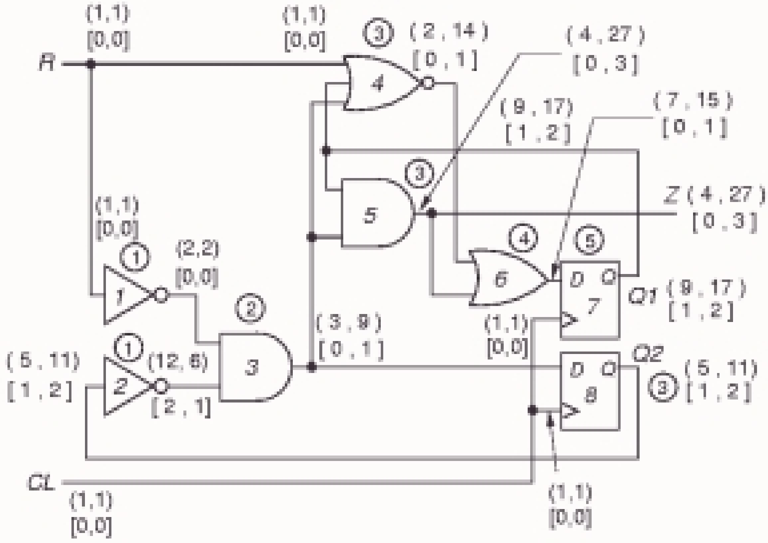
COE – KFUPM Dr. Alaaeldin Amin (COE 545) Slide Number 28

After 2 Iterations



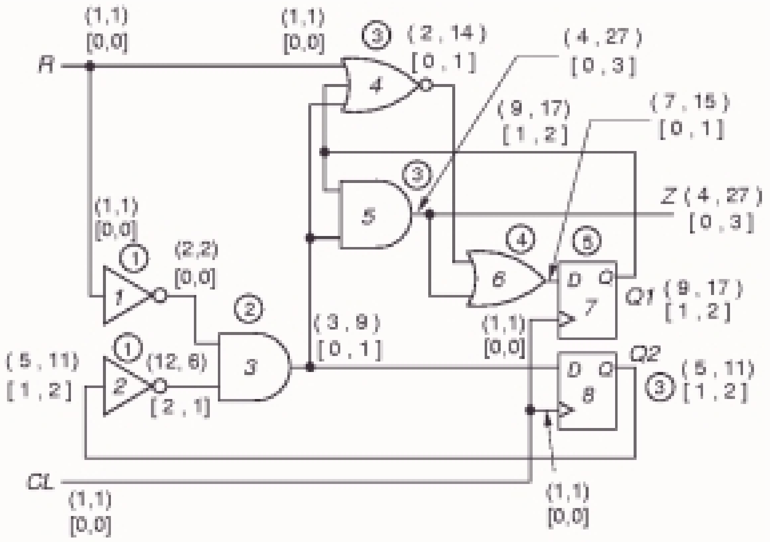
COE – KFUPM | Dr. Alaaeldin Amin (COE 545) | Slide Number 29

After 3 Iterations



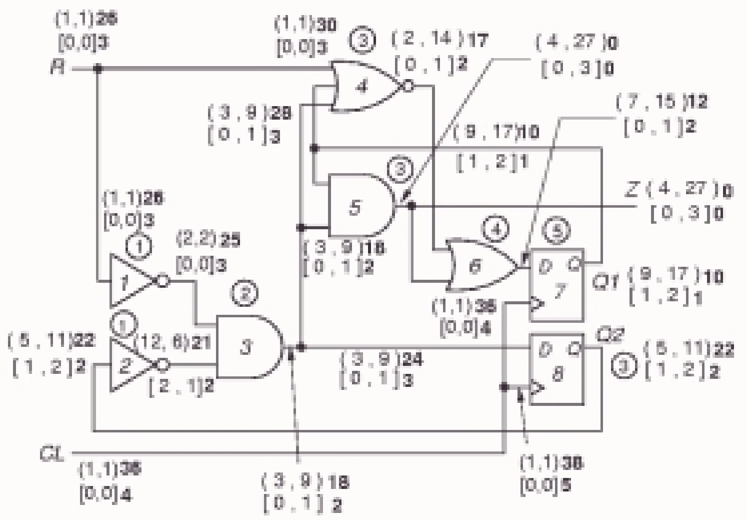
COE – KFUPM | Dr. Alaaeldin Amin (COE 545) | Slide Number 30

Stable Sequential Measures



COE – KFUPM | Dr. Alaaeldin Amin (COE 545) | Slide Number 31

Final Sequential Observabilities

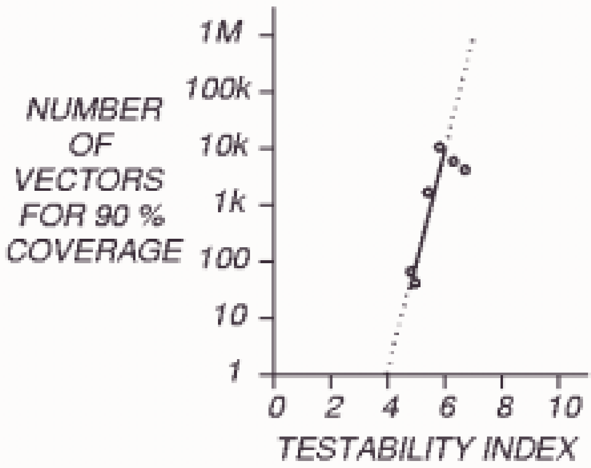


COE – KFUPM | Dr. Alaaeldin Amin (COE 545) | Slide Number 32

Test Vector Length Prediction

- ***Testability*** **for node** ***x*** **stuck-at faults**
  - ➢ T (x_sa_0) = $CC1$ (x) + CO (x)
  - ➢ T (x_sa_1) = $CC0$ (x) + CO (x)
- **Compute** ***testabilities*** **for all stuck-at faults T (f_i)**
- **Define the Testability Index of a Circuit as Follows**
  - ➢ ***Testability index*** = Log $\sum_{\forall f_i} T(f_i)$
- **An Almost Linear Relation was found to Exist between The Testability Index of a Circuit and the Number of TVs for 90% Fault Coverage**

Number Test Vectors vs. Testability Index

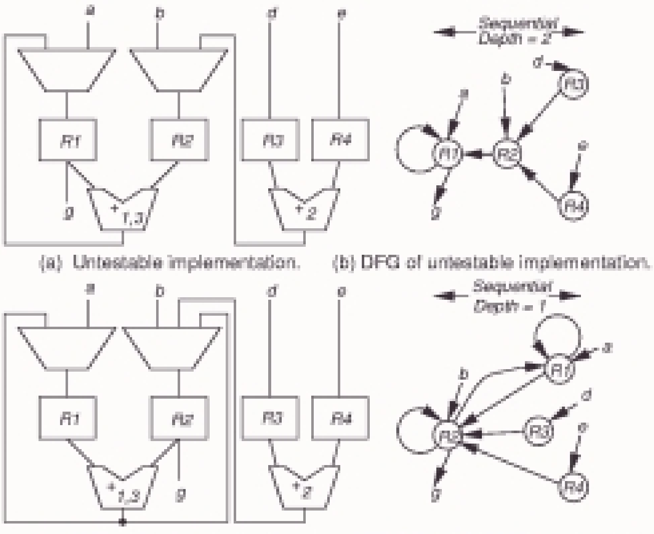


High Level Testability

- Build *data path control graph* (DPCG) for circuit
- Compute *sequential depth* -- # arcs along path between PIs, registers, and POs
- Improve Register Transfer Level Testability with redesign

COE – KFUPM Dr. Alaaeldin Amin (COE 545) Slide Number 35

Improved RTL Design



(a) Untestable implementation. (b) DFG of untestable implementation.

(c) Testable implementation. (d) DFG of testable implementation.

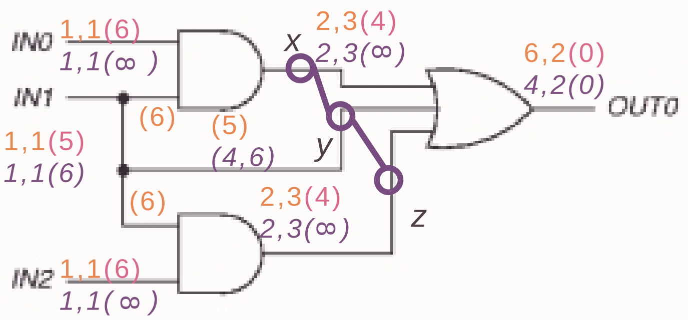
COE – KFUPM Dr. Alaaeldin Amin (COE 545) Slide Number 36

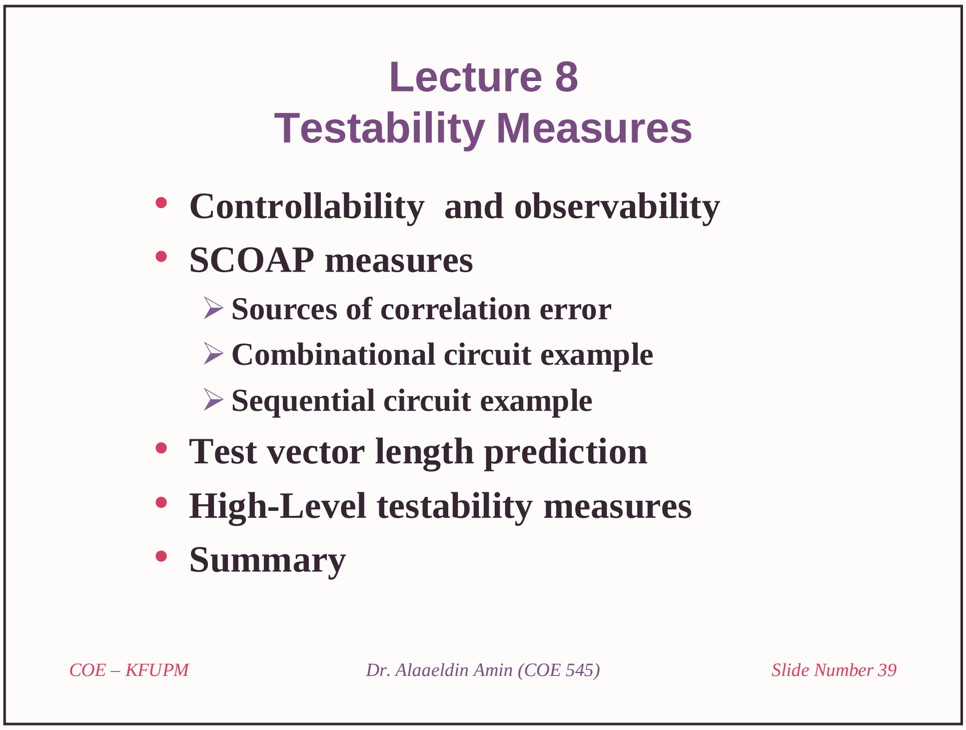
Summary

- **Testability approximately measures:**
  - **Difficulty of setting circuit lines to 0 or 1**
  - **Difficulty of observing internal circuit lines**
- **Uses:**
  - **Analysis of difficulty of testing internal circuit parts**
    - **Redesign circuit hardware or add special test hardware where measures show bad controllability or observability**
  - **Guidance for algorithms computing test patterns – avoid using hard-to-control lines**
  - **Estimation of fault coverage – 3-5 % error**
  - **Estimation of test vector length**

Correlation Error Example

- **Exact computation of measures is NP-Complete and impractical**
- **SCOAP measures are in red or bold CC0,CC1 (CO) -- Italicized (Blue) measures show correct values**




Lecture 8
Testability Measures
Controllability and observability
SCOAP measures
Sources of correlation error
Combinational circuit example
Sequential circuit example
Test vector length prediction
High-Level testability measures
Summary
COE – KFUPM
Dr. Alaaeldin Amin (COE 545)
Slide Number 39