


Testing Digital Systems I

Lecture 9: Boolean Testing Using Fault Models (D-Algorithm, PODEM)

Instructor: M. Tahoori

Copyright 2010, M. Tahoori TDS I: Lecture 9 1



D Algorithm (More Examples)

Copyright 2010, M. Tahoori TDS I: Lecture 9 2

Example: A/0

- Step 1
 - D-Drive: Set A = 1

Copyright 2010, M. Tahoori TDS I: Lecture 9 3

Example: A/0

- Step 2
 - D-Drive : Set f = 0

Copyright 2010, M. Tahoori TDS I: Lecture 9 4

Example: A/0

- Step 3
 - D-Drive : Set $k = 1$

Copyright 2010, M. Tahoori TDS I: Lecture 9 5

Example: A/0

- Step 4
 - Consistency: Set $g = 1$

Copyright 2010, M. Tahoori TDS I: Lecture 9 6

Example: A/0

- Step 5
 - Consistency: $f = 0$
 - Already set

The diagram shows a logic circuit with inputs A, B, C, and D. Input A is marked with a red '1' and a cross, labeled 'sa0'. The circuit consists of several gates: an AND gate with inputs A and B (output e), an OR gate with inputs e and C (output f), an OR gate with inputs f and D (output h), an AND gate with inputs h and D (output k), an OR gate with inputs k and D (output g), and a final AND gate with inputs g and D (output L, which is D-bar). Red annotations indicate the current state: A=1, B=D, C=D, f=0, g=1, k=1, h=D, and L=D-bar.

Copyright 2010, M. Tahoori TDS I: Lecture 9 7

Example: A/0

- Step 6
 - Consistency: Set $c = 0$, Set $e = 0$

The diagram shows the same logic circuit as in Step 5. Red annotations indicate the updated state: A=1, B=D, C=0, e=0, f=0, g=1, k=1, h=D, and L=D-bar.

Copyright 2010, M. Tahoori TDS I: Lecture 9 8

Example: A/0

- Step 7
 - Consistency: Set $B = 0$
- Test found: $ABCD = 100X$

The diagram shows a logic circuit with inputs A, B, C, D and output L. The circuit consists of several gates: an AND gate (A, B) with output e; an OR gate (e, C) with output f; an AND gate (f, D) with output h; an OR gate (h, D) with output k; an AND gate (k, D) with output L. A fault 'sa0' is indicated on input A. The values for each node are: A=1, B=0, C=0, D=X, e=0, f=0, g=1, h=D, k=1, L=D-bar.

Copyright 2010, M. Tahoori TDS I: Lecture 9 9

Example s/1

- Primitive D-cube of Failure

The diagram shows a logic circuit with inputs A, B, C and outputs X, Y, Z. The circuit consists of several gates: an AND gate (A, B) with output d; an OR gate (d, C) with output g; an AND gate (g, C) with output k; an OR gate (k, C) with output m; an AND gate (m, C) with output n; an AND gate (n, C) with output p; an AND gate (p, C) with output q; an AND gate (q, C) with output r; an AND gate (r, C) with output s; an AND gate (s, C) with output u; an AND gate (u, C) with output v; an AND gate (v, C) with output X; an AND gate (X, C) with output Y; an AND gate (Y, C) with output Z. A fault 'sa1' is indicated on node s.

Copyright 2010, M. Tahoori TDS I: Lecture 9 10

Example s/1

- Propagation D-cube for v

Copyright 2010, M. Tahoori TDS I: Lecture 9 11

Example s/1

- Forward & Backward Implications

Copyright 2010, M. Tahoori TDS I: Lecture 9 12

Example s/1

- Propagation D-cube for Z
- test found!

Copyright 2010, M. Tahoori TDS I: Lecture 9 13

Example: u/1

- Primitive D-cube of Failure

Copyright 2010, M. Tahoori TDS I: Lecture 9 14

Example: u/1

- Propagation D-cube for v

Copyright 2010, M. Tahoori TDS I: Lecture 9 15

Example: u/1

- Forward and backward implications

Copyright 2010, M. Tahoori TDS I: Lecture 9 16

Example: u/1


- Propagation D-cube for v

Copyright 2010, M. Tahoori TDS I: Lecture 9 19

Example: u/1

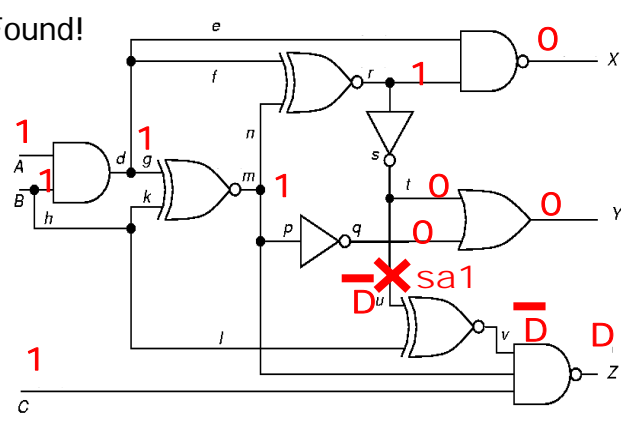
- Propagation D-cube for Z

Copyright 2010, M. Tahoori TDS I: Lecture 9 20




Example: u/1

- Propagation D-cube for Z
- Implications
- Test Found!



Copyright 2010, M. Tahoori
TDS I: Lecture 9
21



PODEM

Copyright 2010, M. Tahoori
TDS I: Lecture 9
22



Motivation

- IBM introduced semiconductor DRAM memory into its mainframes – late 1970's
- Memory had error correction and translation circuits
 - To improved reliability
- D-ALG failed to generate test for these circuits
 - Search too undirected
 - Large XOR-gate trees
 - Must set all external inputs to define output
- Needed a better ATPG tool



PODEM -- Goel IBM (1981)

- Path Oriented DEcision Making
- New concepts introduced:
 - Expand binary decision tree only around primary inputs
 - This reduced size of tree from 2^n to $2^{\text{num_PI}}$
 - Use X-PATH-CHECK
 - To test whether D-frontier still there
 - D-Algorithm tends to continue intersecting D-Cubes
 - Even when D-Frontier disappeared
 - Objectives
 - bring ATPG closer to propagating D (\bar{D}) to PO
 - Backtracing
 - To obtain a PI assignment given an initial objective

Assigning Input Values (PODEM)

1. Assign value to an unassigned primary input
2. Determine all implications of assignment
3. If test is generated, exit; else
4. Is test is possible with additional input assignments ?
 - fault site doesn't have fault value assigned
 - Path of unassigned leads from D (\bar{D}) to an output
 - If yes, go to 1, if no
5. Change input assignments to untried combination, go to 2
 - If no untried combination exists — untestable fault

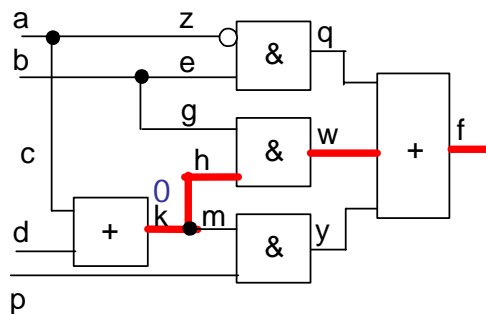
Copyright 2010, M. Tahoori

TDS I: Lecture 9

25

Example: Test For k/1

- Put D' on k
 - D-Alg: assigned a D' to k and propagate it to output f
 - PODEM: try to justify 0 on k



Copyright 2010, M. Tahoori

TDS I: Lecture 9

26

Example: Test For k/1

- Justify 0 on d
- Implication

Copyright 2010, M. Tahoori TDS I: Lecture 9 27

Example: Test For k/1

- K still hasn't D'
- Justify 0 on c
- Implication: $k=h=m=D'$

Copyright 2010, M. Tahoori TDS I: Lecture 9 28

Example: Test For k/1


- Propagate through w
- Set $g = 1$
- Implication

Copyright 2010, M. Tahoori TDS I: Lecture 9 29

Example: Test For k/1

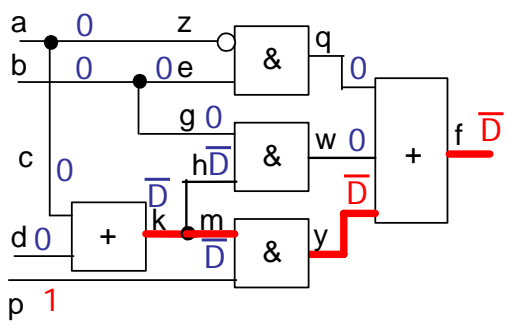
- Conflict
 - f is 1 so propagation is blocked
- Reverse the last assignment made to a PI
 - Set $b = 0$
- Implication

Copyright 2010, M. Tahoori TDS I: Lecture 9 30



Example: Test For k/1


- There is a propagation path from m to f
 - Set $p = 1$
- Implication
- Test found
 - $abcd = 0001$



Copyright 2010, M. Tahoori

TDS I: Lecture 9

31



Another Example

Copyright 2010, M. Tahoori

TDS I: Lecture 9

32

Example: S/1

- Select path s – Y for fault propagation

The diagram shows a logic circuit with three inputs: A, B, and C. The circuit consists of several gates: an AND gate (d, g), an OR gate (k, m), an AND gate (e, f), an OR gate (r, s), an AND gate (p, q), an OR gate (u, v), and an AND gate (i, Z). A path from node s to output Y is highlighted in red. A green 'sa1' label is placed near node s.

Copyright 2010, M. Tahoori TDS I: Lecture 9 33

Example: S/1

- Initial objective:
 - Set r to 1 to sensitize fault

The diagram shows the same logic circuit as in the previous slide. A path from node s to output Y is highlighted in red. A red circle with the number 7 is placed near node r. A blue 'sa1' label is placed near node s.

Copyright 2010, M. Tahoori TDS I: Lecture 9 34

Example: S/1

- Backtrace from r

Copyright 2010, M. Tahoori TDS I: Lecture 9 35

Example: S/1

- Set A = 0 in implication stack

Copyright 2010, M. Tahoori TDS I: Lecture 9 36

Example: S/1

- Forward implications: $d = 0, X = 1$

Copyright 2010, M. Tahoori TDS I: Lecture 9 37

Example: S/1

- Initial objective: set r to 1

Copyright 2010, M. Tahoori TDS I: Lecture 9 38

Example: S/1

- Backtrace from r again

Copyright 2010, M. Tahoori TDS I: Lecture 9 39

Example: S/1

- Set B to 1.
 - Implications in stack: $A = 0, B = 1$

Copyright 2010, M. Tahoori TDS I: Lecture 9 40

Example: S/1

- Forward implications:
 - $k = 1, m = 0, r = 1, q = 1, Y = 1, s = \bar{D}, u = \bar{D}, v = \bar{D}, Z = 1$

Copyright 2010, M. Tahoori TDS I: Lecture 9 41

Example: S/1

- X-PATH-CHECK
 - paths $s - Y$ and $s - u - v - Z$ blocked

Copyright 2010, M. Tahoori TDS I: Lecture 9 42

Example: S/1

- Set B = 0 (alternate assignment)

Copyright 2010, M. Tahoori TDS I: Lecture 9 43

Example: S/1

- Forward implications:
- $d = 0, X=1, m = 1, r = 0, s = 1, q = 0, Y = 1, v = 0, Z = 1$
- Fault not sensitized

Copyright 2010, M. Tahoori TDS I: Lecture 9 44

Example: S/1

- Set A = 1 (alternate assignment)

Copyright 2010, M. Tahoori TDS I: Lecture 9 45

Example: S/1

- Backtrace from r again

Copyright 2010, M. Tahoori TDS I: Lecture 9 46

Example: S/1

- Set $B = 0$.
- Implications in stack: $A = 1, B = 0$

Copyright 2010, M. Tahoori TDS I: Lecture 9 47

Example: S/1

- Forward implications:
 - $d = 0, X = 1, m = 1, r = 0$.
- Conflict: fault not sensitized. Backtrack

Copyright 2010, M. Tahoori TDS I: Lecture 9 48

Example: S/1

- Set B = 1 (alternative assignment)

Copyright 2010, M. Tahoori TDS I: Lecture 9 49

Example: S/1

- Forward implications:
 - $d = 1, m = 1, r = 1, q = 0, s = \bar{D}, v = \bar{D}, X = 0, Y = \bar{D}$
- Test found

Copyright 2010, M. Tahoori TDS I: Lecture 9 50




PODEM

- Major aspects
 - Which primary input should be assigned a logic value?
 - What value to assign to the selected primary input?
 - Determining inconsistencies in primary input assignments
 - Handling inconsistencies



Which PI to Choose?


- Decision gate
 - Logic value at the output of a gate is such that only one input of the gate can control its output to the desired value
 - AND with output 0
- Imply gate
 - Logic value at the output of a gate is such that all inputs of the gate must be at a particular value in order to control its output to the desired value
 - AND with output 1
- To justify a logic value at the output of a decision gate, choose the "easiest" input.
 - The shortest logical path to primary inputs or has the best controllability
- To justify a logic value at the output of an imply gate, choose the "hardest" input
 - The longest logical path to primary inputs or has the worst controllability



What Value to Assign?

- Path from the objective site to the selected primary input has an **even** number of inversions
 - Assign the same value to PI as the objective
- Path from the objective site to the selected primary input has an **odd** number of inversions
 - Assign the opposite value of the objective to PI

Copyright 2010, M. Tahoori TDS I: Lecture 9 53



Inconsistencies in PI Assignment

- After every primary input assignment, an implication step is performed.
- During implication, inconsistencies in primary input assignments are detected using the following rules:
 - If there are conflicting assignments at the same signal line of the network
 - If the logic value at the fault site doesn't activate the fault
 - If there is no path from the fault site to a primary output such that all side inputs of that path are either X or set at non-controlling values

Copyright 2010, M. Tahoori TDS I: Lecture 9 54



Handling Inconsistencies

- Backtracking
 - Flip the logic value at the primary input
 - Which was the last one to be assigned a value
 - Stack of primary inputs that have been assigned values
 - After flipping implication step is performed
 - No inconsistency detected
 - Continue
 - Otherwise
 - That primary input is removed from the stack and
 - X is assigned to that primary input
 - POP the next assigned PI from stack and repeat