EE434
ASIC & Digital Systems

Testing for Single-Stuck Faults

Spring 2016
Dae Hyun Kim
daehyun@eecs.wsu.edu

Definition of Automatic Test-Pattern Generator (ATPG)

- **Operations on digital hardware:**
  - **Inject fault into circuit modeled in computer**
  - **Use various ways to activate and propagate fault effect through hardware to circuit output**
  - **Output flips from expected to faulty signal**
- ***Electron-beam* (*E-beam*) *test* observes internal signals – "picture" of nodes charged to 0 and 1 in different colors**
  - **Too expensive**
- ***Scan design* – add test hardware to all flip-flops to make them a giant shift register in test mode**
  - **Can shift state in, scan state out**
  - **Widely used – makes sequential test combinational**
  - **Costs: 5 to 20% chip area, circuit delay, extra pin, longer test sequence**

Test Generation

- Controlling value c
  - determines the value of the gate output regardless of the values of the other inputs.
  - A control value on an input of a gate blocks propagation of faults from other inputs.
- Inversion i
  - Inversion value of a gate is 0 if no inversion is done (otherwise 1).
  - The output value is $c \oplus i$.

	c	i
AND	0	0
OR	1	0
NAND	0	1
NOR	1	1

Test Generation

- Composite logic values & 5-valued operations

v/v_f	
0/0	0
1/1	1
1/0	D
0/1	$\overline{D}$

AND	0	1	D	$\overline{D}$	x
0	0	0	0	0	0
1	0	1	D	$\overline{D}$	x
D	0	D	D	0	x
$\overline{D}$	0	$\overline{D}$	0	$\overline{D}$	x
x	0	x	x	x	x

Test Generation

- Composite logic values & 5-valued operations

v/v_f	
0/0	0
1/1	1
1/0	D
0/1	$\overline{D}$

OR	0	1	D	$\overline{D}$	x
0	0	1	D	$\overline{D}$	x
1	1	1	1	1	1
D	D	1	D	1	x
$\overline{D}$	$\overline{D}$	1	1	$\overline{D}$	x
x	x	1	x	x	x

Test Generation

- Generate a test for l *s-a-**v***.

set all values to x

Justify (l, $\bar{v}$)

if $v = 0$, **then**

  Propagate (l, D)

else

  Propagate (l, $\bar{D}$)

Justify (l, v)

  set l to v

  if l is a PI, **then return**

  // now l is a gate (output)

  c = controlling value of l

  i = inversion of l

  $inval$ = $v \oplus i$

  if $inval = \bar{c}$, **then**

    for every input j of l

      Justify (j, $inval$)

  else

    select one input (j) of l

    Justify (j, $inval$)

Propagate (l, err)

  set l to err

  if l is a PO, **then return**

  k = the fanout of l

  c = controlling value of k

  i = inversion of k

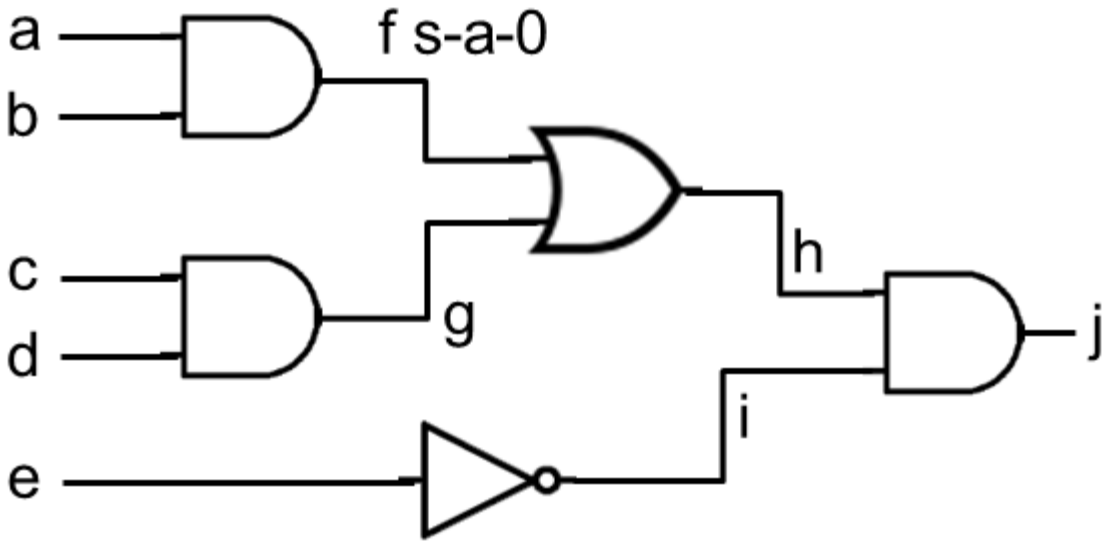
  for every input j of k other than l

    Justify (j, $\bar{c}$)

  Propagate (k, $err \oplus i$)

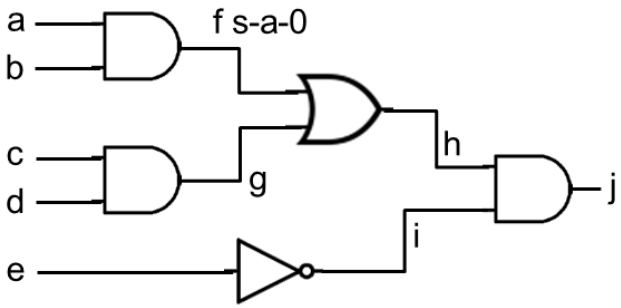
Test Generation

- Example



Test Generation

- Example



set all values to x
Justify (l, $\bar{v}$)
if $v = 0$, **then**
  Propagate (l, D)
else
  Propagate (l, $\bar{D}$)

Justify (l, v)
  set l to v
  if l is a PI, **then return**
  c = controlling value of l
  i = inversion of l
  $inval = v \oplus i$
  if $inval = \bar{c}$, **then**
    for every input j of l
      Justify (j, $inval$)
  else
    select one input (j) of l
    Justify (j, $inval$)

Propagate (l, err)
  set l to err
  if l is a PO, **then return**
  k = the fanout of l
  c = controlling value of k
  i = inversion of k
  for every input j of $k (\neq l)$
    Justify (j, $\bar{c}$)
  Propagate (k, $err \oplus i$)

1.
l s-a-v
$l = f$, $v = 0$
Justify (f, 1)
Propagate (f, D)

2.
Justify (f, 1)
$f = 1$
$c = 0$
$i = 0$
$inval = 1 \oplus 0 = 1$
Justify (a, 1)
Justify (b, 1)

3.
Justify (a, 1)
$\boldsymbol{a = 1}$

4.
Justify (b, 1)
$\boldsymbol{b = 1}$

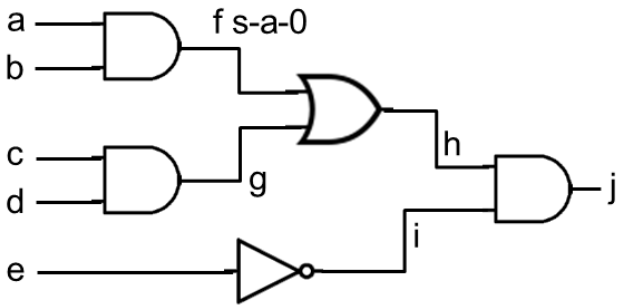
5.
Propagate (f, D)
$f = D$
k = the OR gate
$c = 1$
$i = 0$
Justify (g, 0)
Propagate (h, D)

6.
Justify (g, 0)
$g = 0$
$c = 0$
$i = 0$
$inval = 0 \oplus 0 = 0$
Justify (c, 0)

7.
Justify (c, 0)
$\boldsymbol{c = 0}$

Test Generation

- Example



8.

Propagate (h, D)

$h = D$

$k = j$

$c = 0$

$i = 0$

Justify (i, 1)

Propagate (j, D)

9.

Justify (i, 1)

$i = 1$

=> $\boldsymbol{e = 0}$

10.

Propagate (j, D)

$j = D$

set all values to x

Justify (l, $\bar{v}$)

if $v = 0$, **then**

Propagate (l, D)

else

Propagate (l, $\bar{D}$)

Justify (l, v)

set l to v

if l is a PI, **then return**

c = controlling value of l

i = inversion of l

$inval = v \oplus i$

if $inval = \bar{c}$, **then**

for every input j of l

Justify (j, $inval$)

else

select one input (j) of l

Justify (j, $inval$)

Propagate (l, err)

set l to err

if l is a PO, **then return**

k = the fanout of l

c = controlling value of k

i = inversion of k

for every input j of $k(\neq l)$

Justify (j, $\bar{c}$)

Propagate (k, $err \oplus i$)

abcde = 110x0

$$Z = (ab + cd)\bar{e}$$

$$Z \oplus Z_f = \{(ab + cd)\bar{e}\} \oplus \{cd\bar{e}\} \Longrightarrow abcde = 110x0 \text{ or } abcde = 11x00$$