Logic Optimization

Heuristic Based

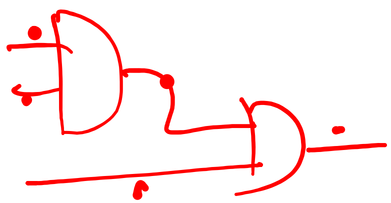
LOGIC
to 2level
verificati
Testing
(TG.)

Virendra Singh
Professor
Department of Electrical Engineering &
Dept. of Computer Science & Engineering
Indian Institute of Technology Bombay
http://www.ee.iitb.ac.in/~viren/
E-mail: viren@{ee, cse}.iitb.ac.in

EE-677: Foundations of VLSI CAD

Logic Testing Based Minimization

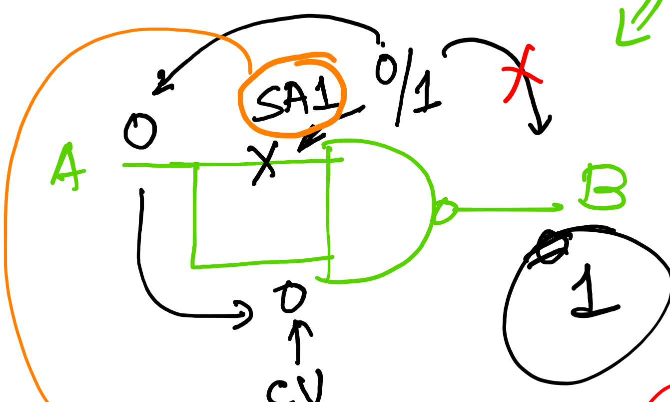
3 SAT
⇑
NP Complete

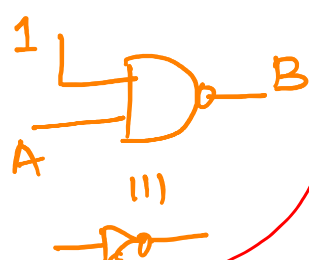


faults = 2 × #nets
= 2 × 5 = 10

PODEM ⇒ { builds a decision tree at PI

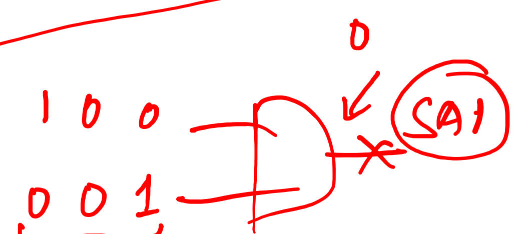
– if we can generate TV for all 10 faults ⇑.

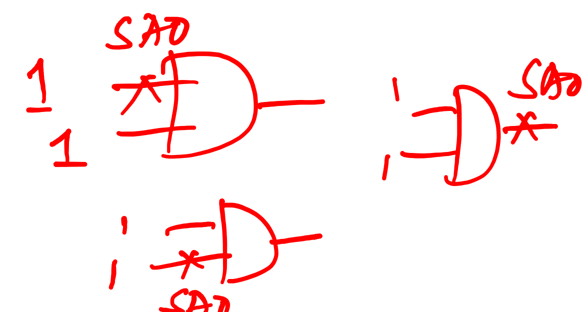




→ One TV can detect multiple faults &

→ One fault can be tested by multiple TVs.





Untestable fault

Redundant fault

Logic Testing Based Minimization

Either we generate a TV for a fault or it declare this fault as redundant

Identification of redundant fault.

Boolean Diff:

⇓

SAT / PODEM ✓

1000

REDUNDANCY

950 ✓ TV

50 → Red.

$FC = \frac{950}{1000} \times 100 = 95\%$

✓ FE $= \frac{950}{1000-50} \times 100 = 100\%$

TC

Test coverage

⇓

$FC = \frac{\#\text{detectable fault}}{\#\text{Total fault}}$

Fault coverage

F 1000, 950 ✓ $\frac{40 \text{ red}}{10}$.

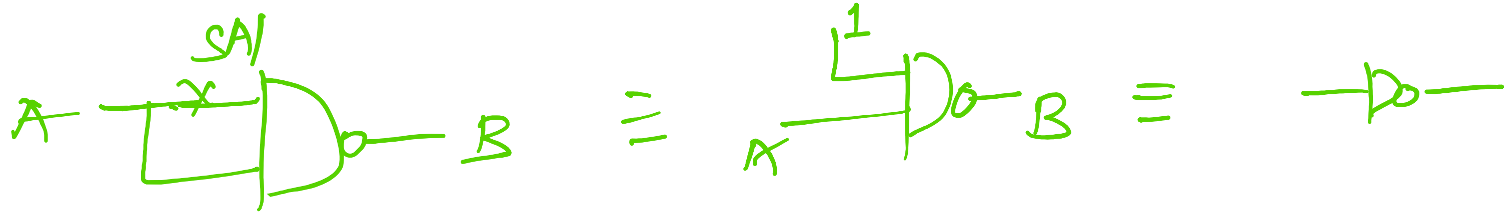
Fault efficiency → FE

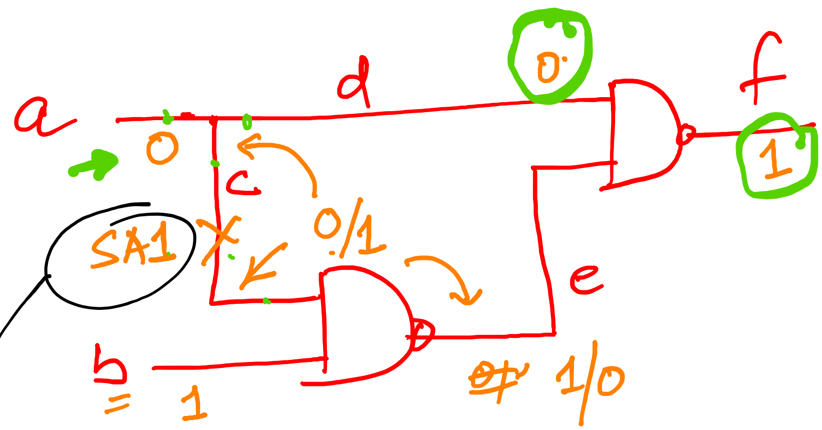
Fault Efficiency $= \frac{\#\text{detectable}}{\#\text{Total faults} - \#\text{red. faults}}$

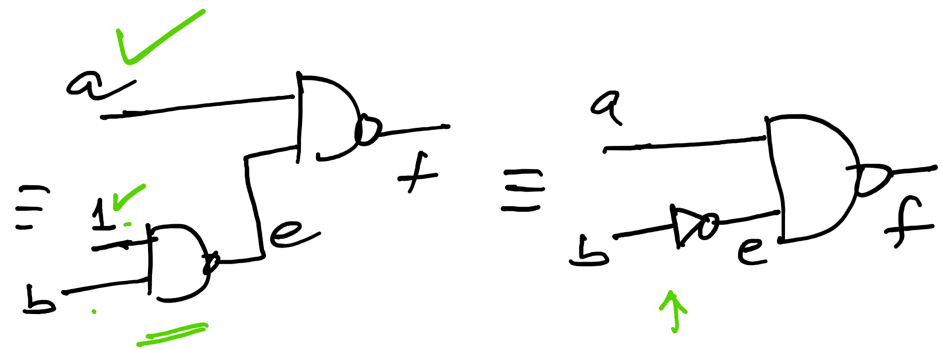
$FC = \frac{950}{1000} \times 100$

$FE = \frac{950}{1000-40} = \frac{950 \times 100}{960}$

Logic Testing Based Minimization





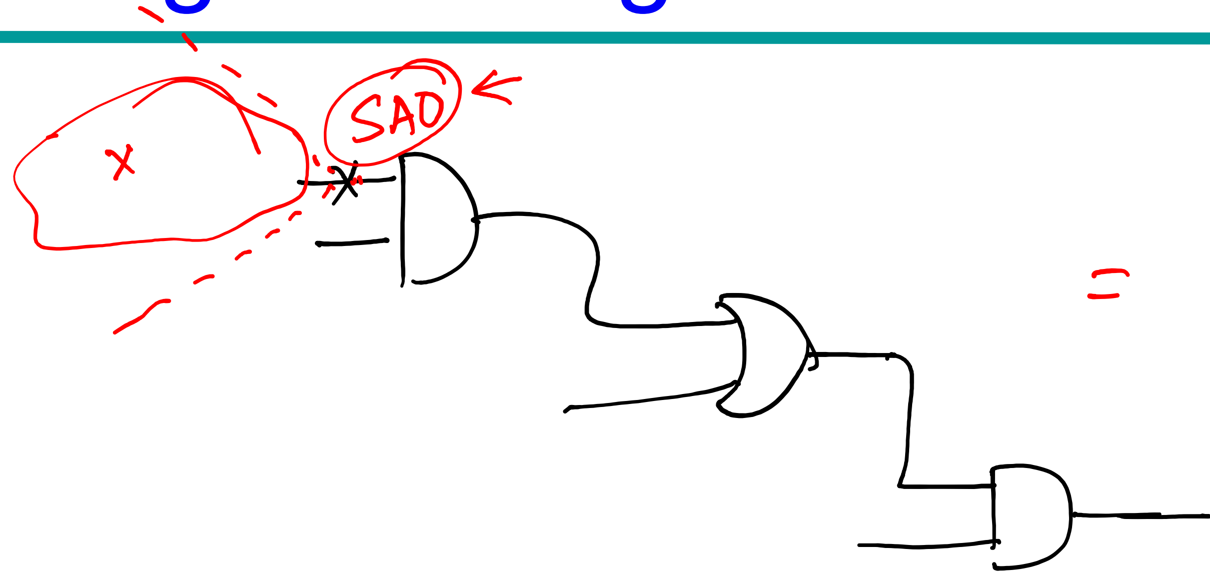


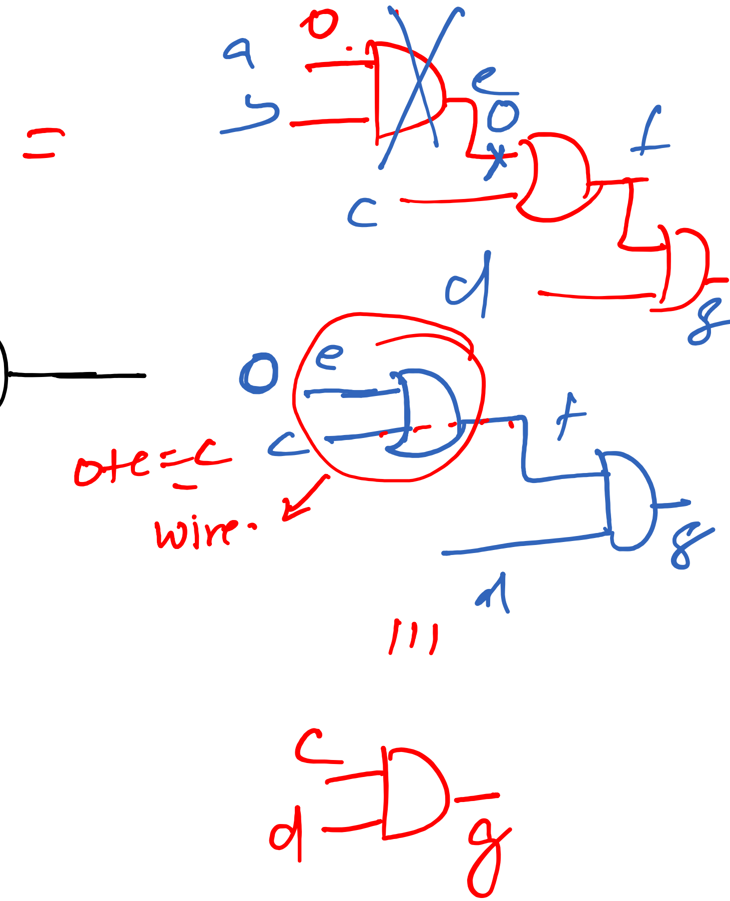
No test vector

Redundant fault

FC & FE

Logic Testing Based Minimization





Redundancy Removal

Heuristic based 2 Level Logic Minimization

Exact Method → QM algo. → all PIs ✓ (EPI) ✓

– Petrick's method

→ ILP

Some more background

- Function $f\ (\ x_1,\ x_2,\ ....,\ x_i,\ ....,\ x_n)$
- Cofactor of f with respect to variable x_i
  - $f_{xi} = f\ (\ x_1,\ x_2,\ ....,\ 1,\ ....,\ x_n)$
- Cofactor of f with respect to variable x_i'
  - $f_{xi'} = f\ (\ x_1,\ x_2,\ ....,\ 0,\ ....,\ x_n)$
- Boole's expansion theorem:
  - $f\ (\ x_1,\ x_2,\ ....,\ x_i,\ ....,\ x_n) = x_i\ fx_i + x_{i'}\ fx_i'$
  - Also credited to Claude Shannon

Example

- Function: $f = ab + bc + ac$
- Cofactors:
  - $f_a = b + c$
  - $f_{a'} = bc$
- Expansion:
  - $f = a\, f_a + a' f_{a'} = a(b + c) + a'bc$

Unateness

RM → n+4

$x_1x_2+x_2x_3+x_ix_4$ positive

UNATE

negative $\bar{x}_1\bar{x}_2+\bar{x}_1\bar{x}_3+\bar{x}_2\bar{x}_4$

- Function $f\,(x_1, x_2, \ldots, x_i, \ldots, x_n)$
- *Positive unate* in x_i when:
  - $f_{x_i} \geq f_{x_i'}$
- *Negative unate* in x_i when:
  - $f_{x_i} \leq f_{x_i'}$
- A function is positive/negative unate when positive/negative unate in all its variables

x_i x_1

$x_1x_2+x_1\bar{x}_3+\bar{x}_2x_3$

$f_{x_1} = x_2+\bar{x}_3+\bar{x}_2x_3$

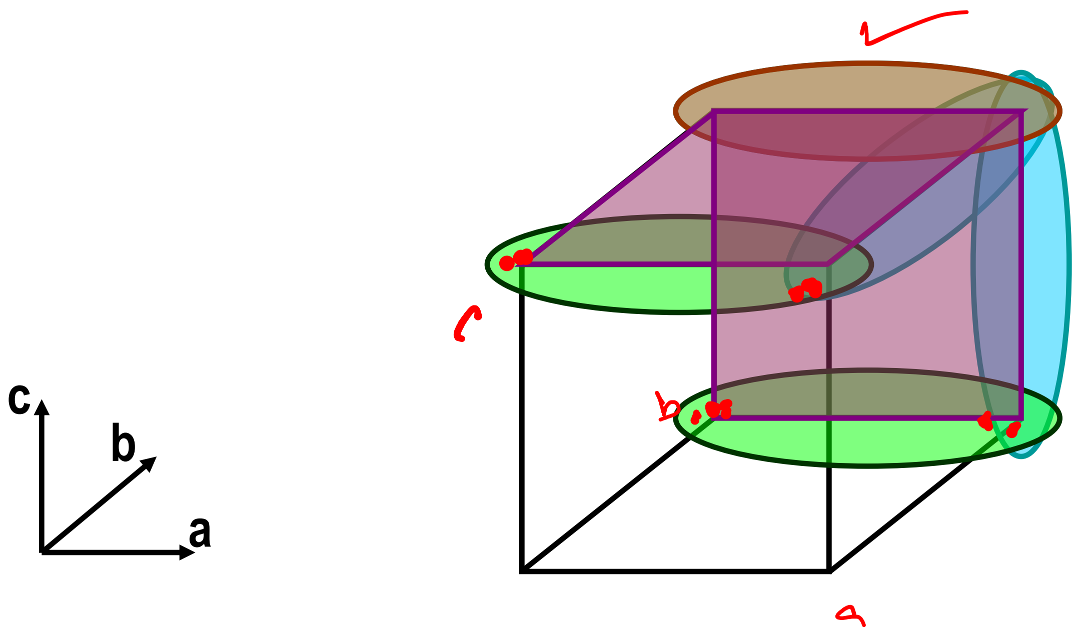
$f_{\bar{x}_1} = \bar{x}_2x_3$

Operators

- Function $f\,(\,x_1,\,x_2,\,\ldots.,\,x_i,\,\ldots.,\,x_n)$
- *Boolean difference* of f w.r.t. variable x_i:
  - $\partial f/\partial x_i \equiv f_{xi} \oplus f_{xi'}$
- *Consensus* of f w.r.t. variable x_i:
  - $C_{xi} \equiv f_{xi} \cdot f_{xi'}$
- *Smoothing* of f w.r.t. variable x_i:
  - $S_{xi} \equiv f_{xi} + f_{xi'}$

Example
$f = ab + bc + ac$

- The Boolean difference $\partial f/\partial a = f_a \oplus f_{a'} = b'c + bc'$
- The consensus $C_a = f_a \cdot f_{a'} = bc$
- The smoothing $S_a \equiv f_a + f_{a'} = b + c$



BEBQUT

Thank You