Logic Testing

ATPG

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

Lecture 24 on 05 Oct 2021

CADSL

Q Algebraic → Boolean Difference
→ SAT.

⇒ [Algorithmic → PODEM. (1981)
Prabhu Goel.
(Verilog)

→ FAN
↓
learning

a = 0

d = 0.

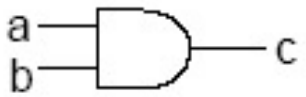
ATPG - Algorithmic

- Path Sensitization Method
  - Fault Sensitization
  - Fault Propagation
  - Line Justification

- Path Sensitization Algorithms
  - D- Algorithm (Roth)
  - PODEM (P. Goel)
  - FAN (Fujiwara)
  - SOCRATES (Schultz)
  - SPIRIT (Emil & Fujiwara)

Common Concept

- Fault Activation problem → a LJ Problem
- The Fault Propagation problem →
  1. Select a FP path to PO → Decision
  2. Once the path is selected → a set of LJ problems
- The LJ Problems → Decisions or Implications



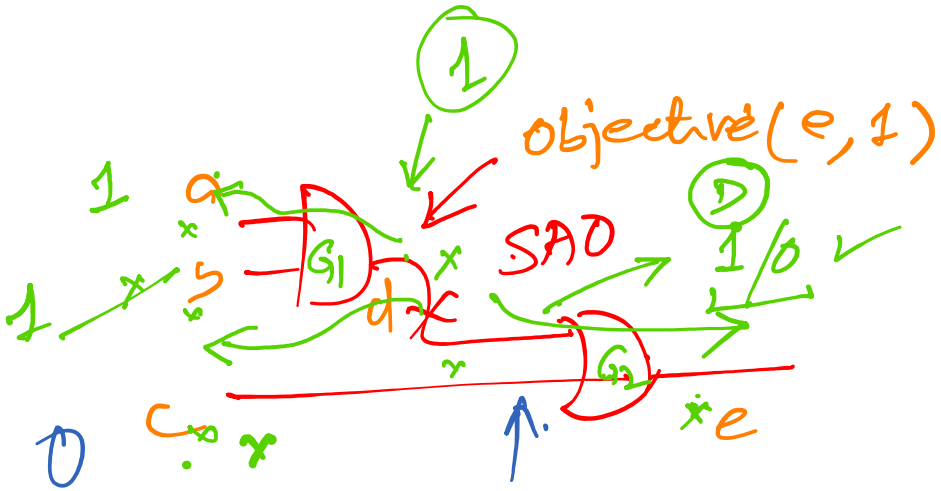
To justify c = 1 → a = 1, b = 1 (Implication)

To justify c = 0 → a = 0 or b = 0 (Decision)

Redundant
UNTESTABLE.

- Incorrect decision → Backtrack → Another decision

Automatic Test Pattern Generation: Algorithmic



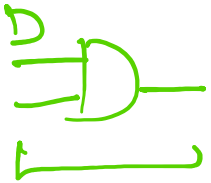
SA0 → D / $\bar{D}$ – SA1

Path Oriented Decision Making

Objective(e, 0)

ncv.

Fanout free ⇒ always you can get a test vector.



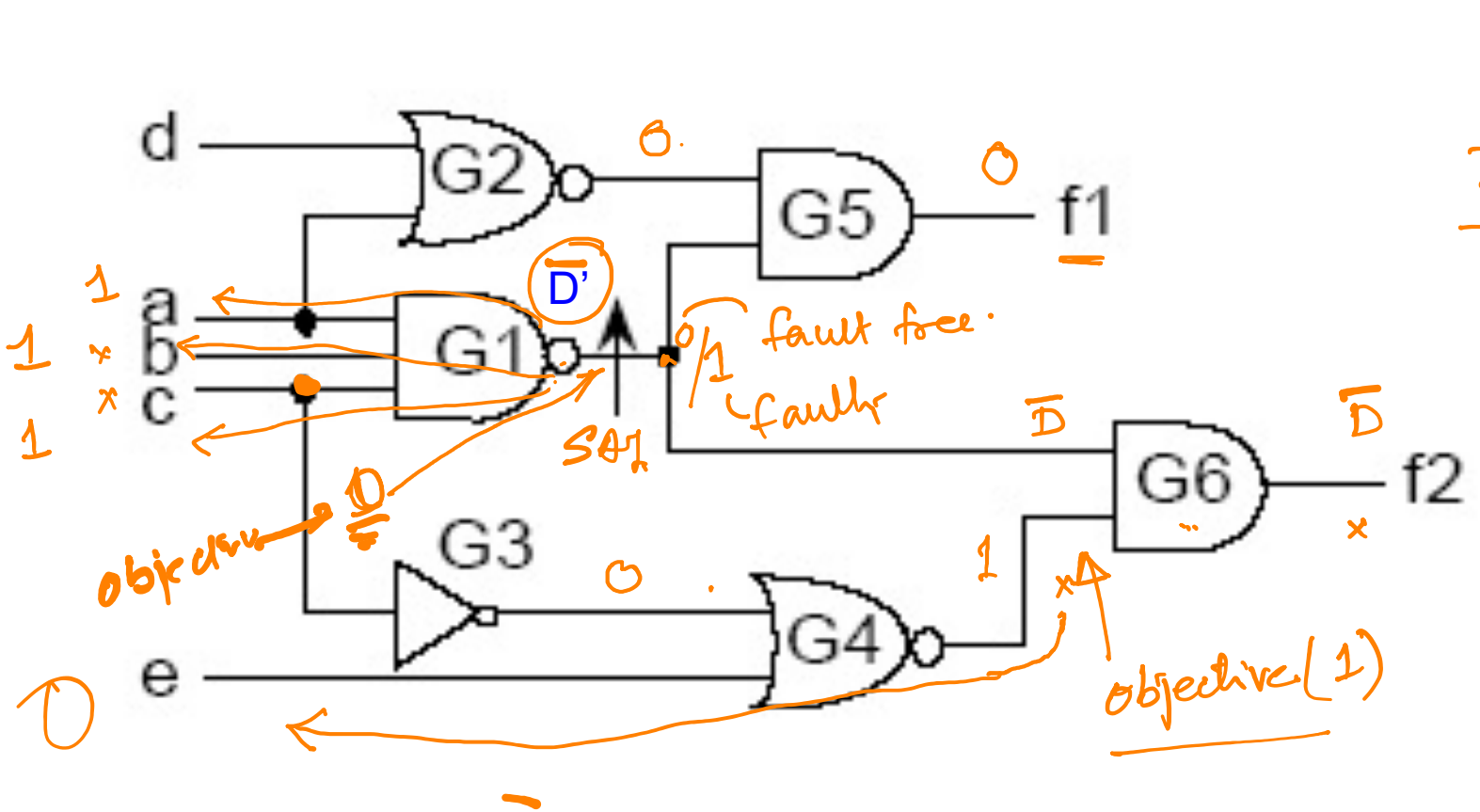
D-frontier list

G_2

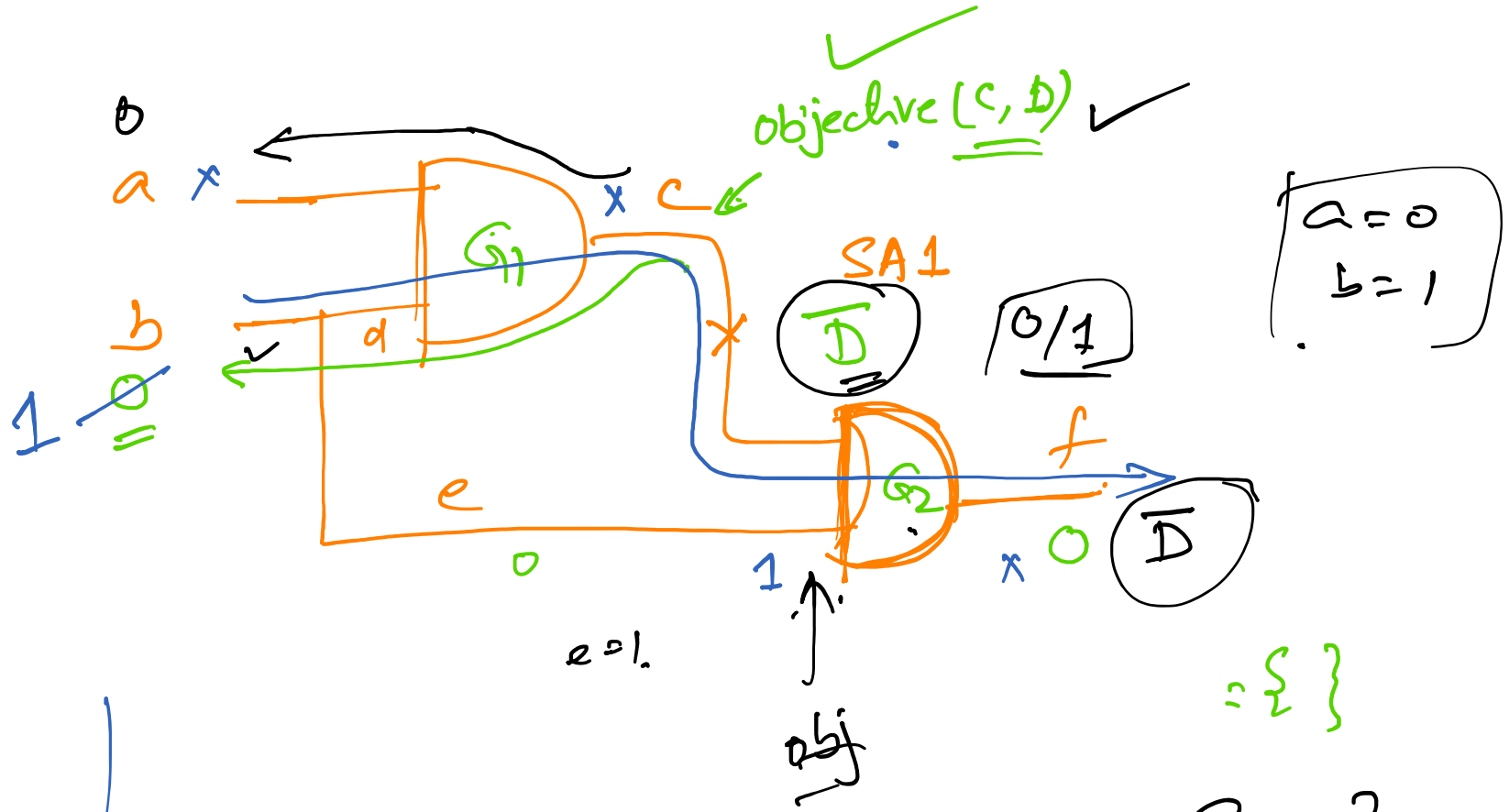
$\{G_2\}$

Fanout → Reconvergent.

PODEM



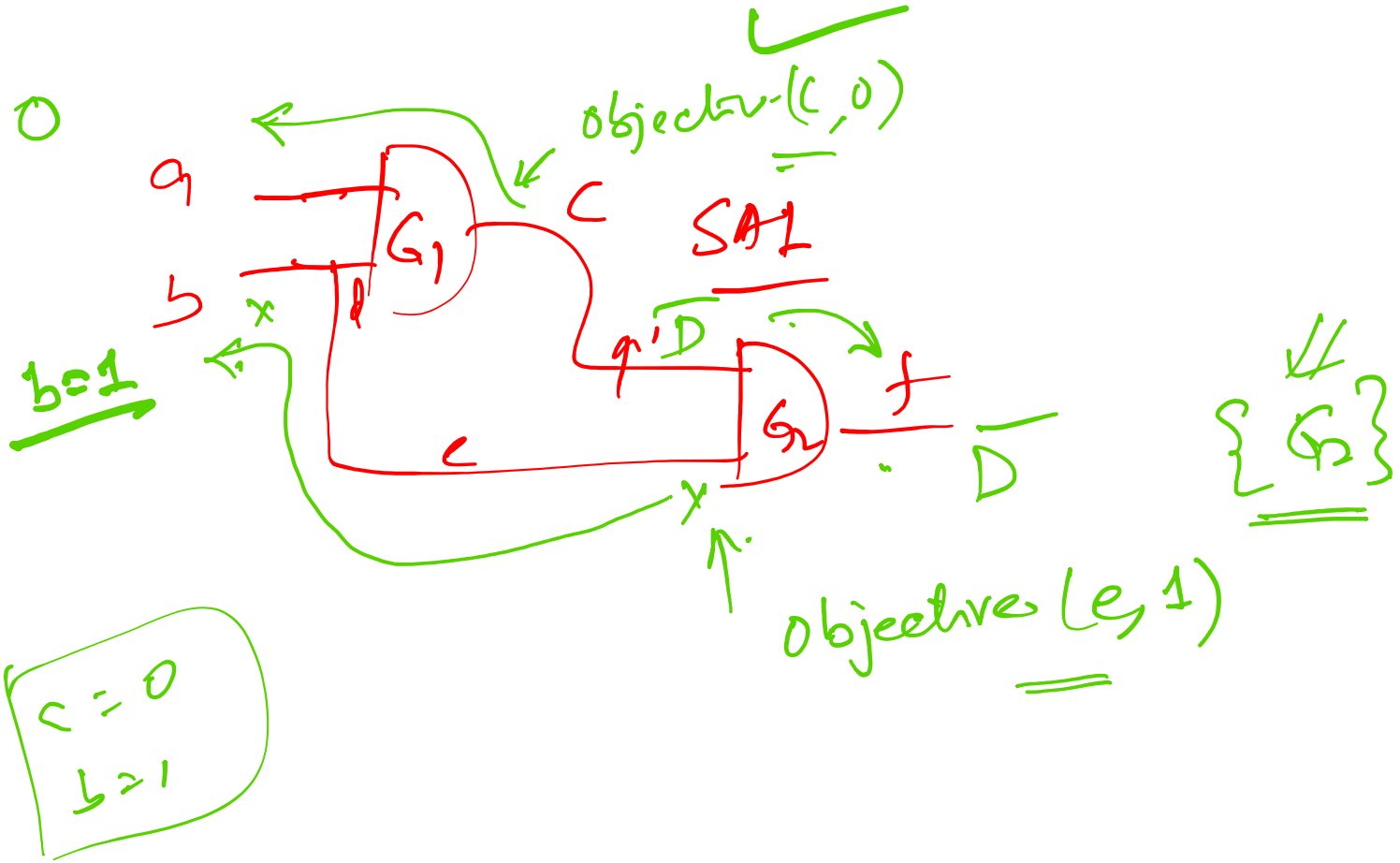
D – frontier: The set of all gates whose output value is currently x but have one or more fault signals on their inputs

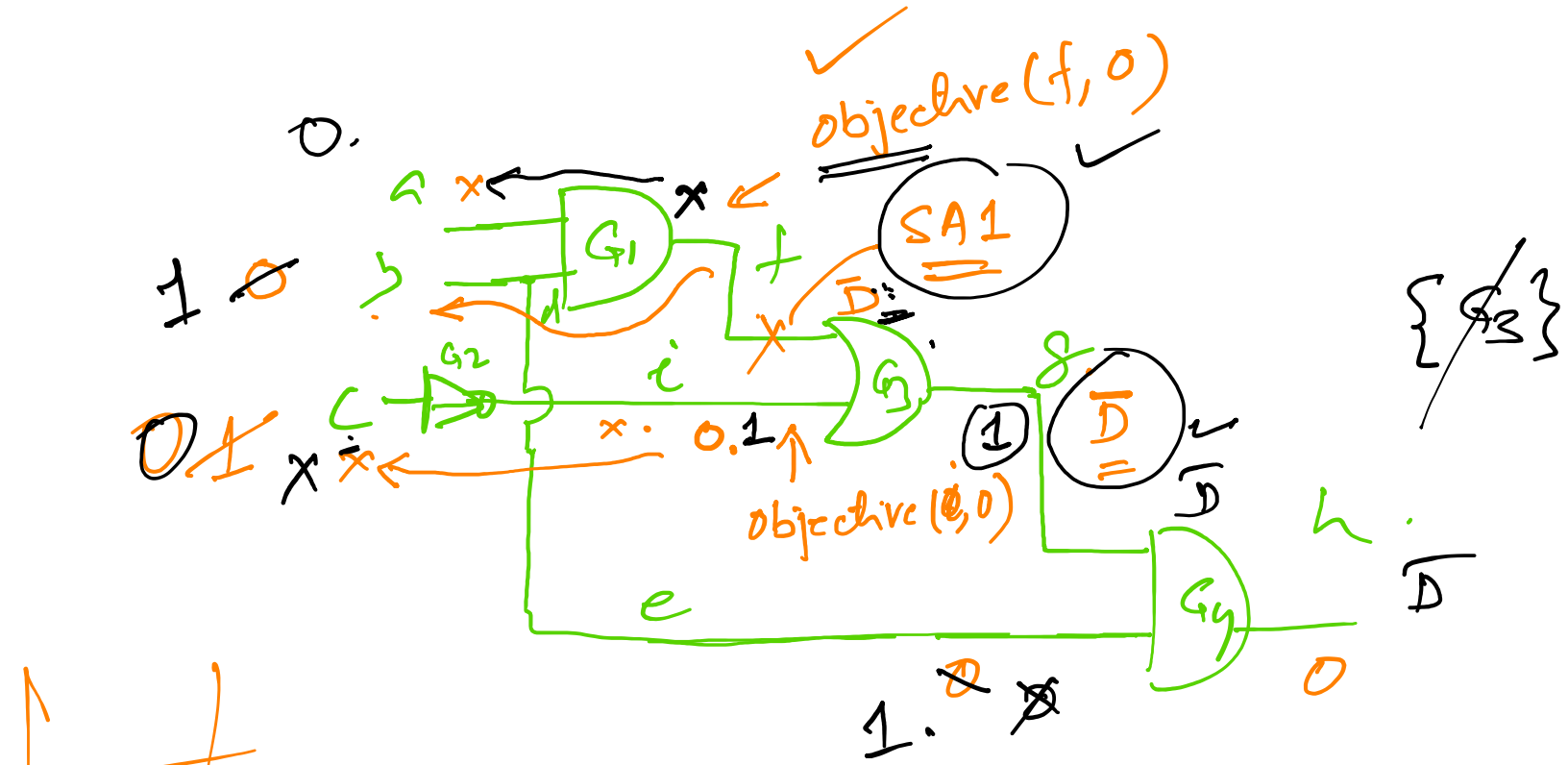


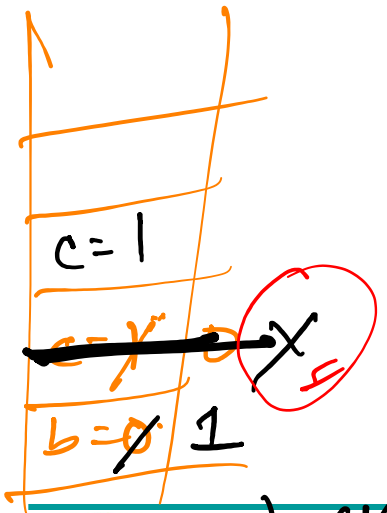
a = b

b = ~~0~~ 1 ← backtrack

Steps of decisions


0
objective (c, 0)
a
b
x
G1
c
SAT
q'D
G2
f
D
{G2}
b=1
x
objective (c, 1)
c = 0
b = 1


objective (f, 0)
SA1
G1
G2
G3
G4
objective (e, 0)
a
b
c
d
e
f
g
h
i
D
1
0


c = 1
b = 0 1

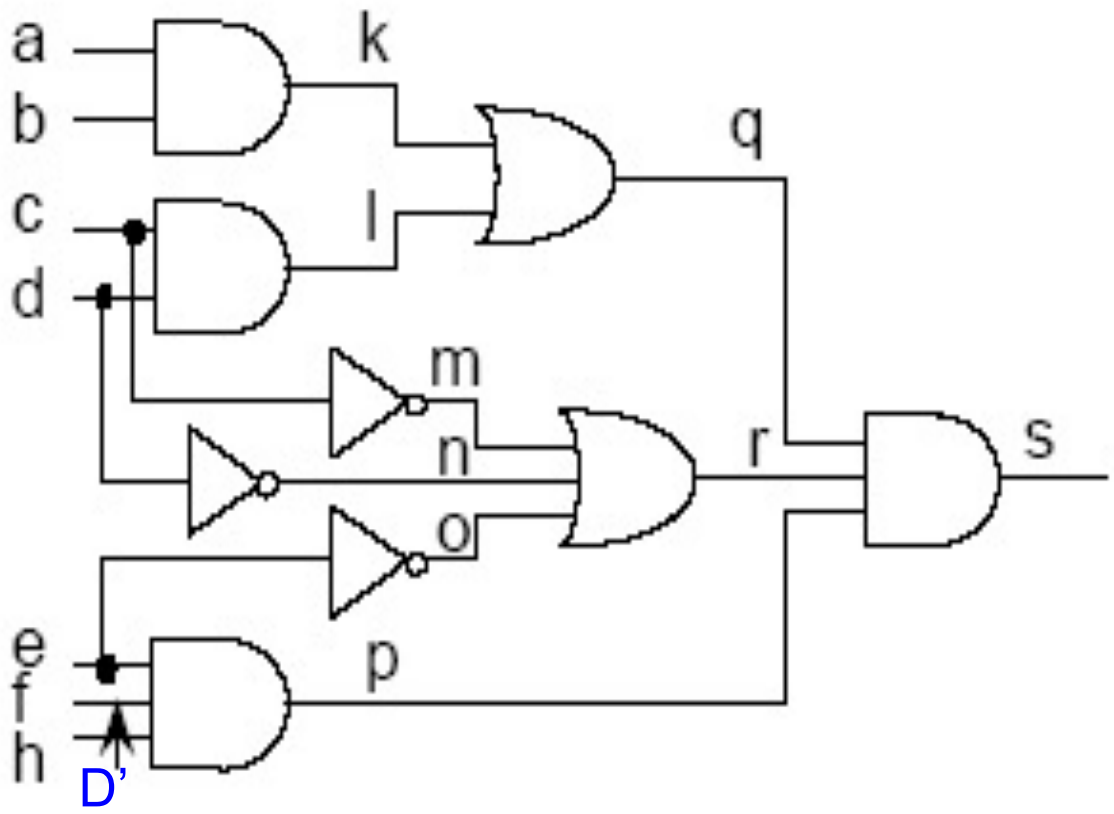
a var :

{ G3 }

⇓ { } X

G3

Decisions during LJ

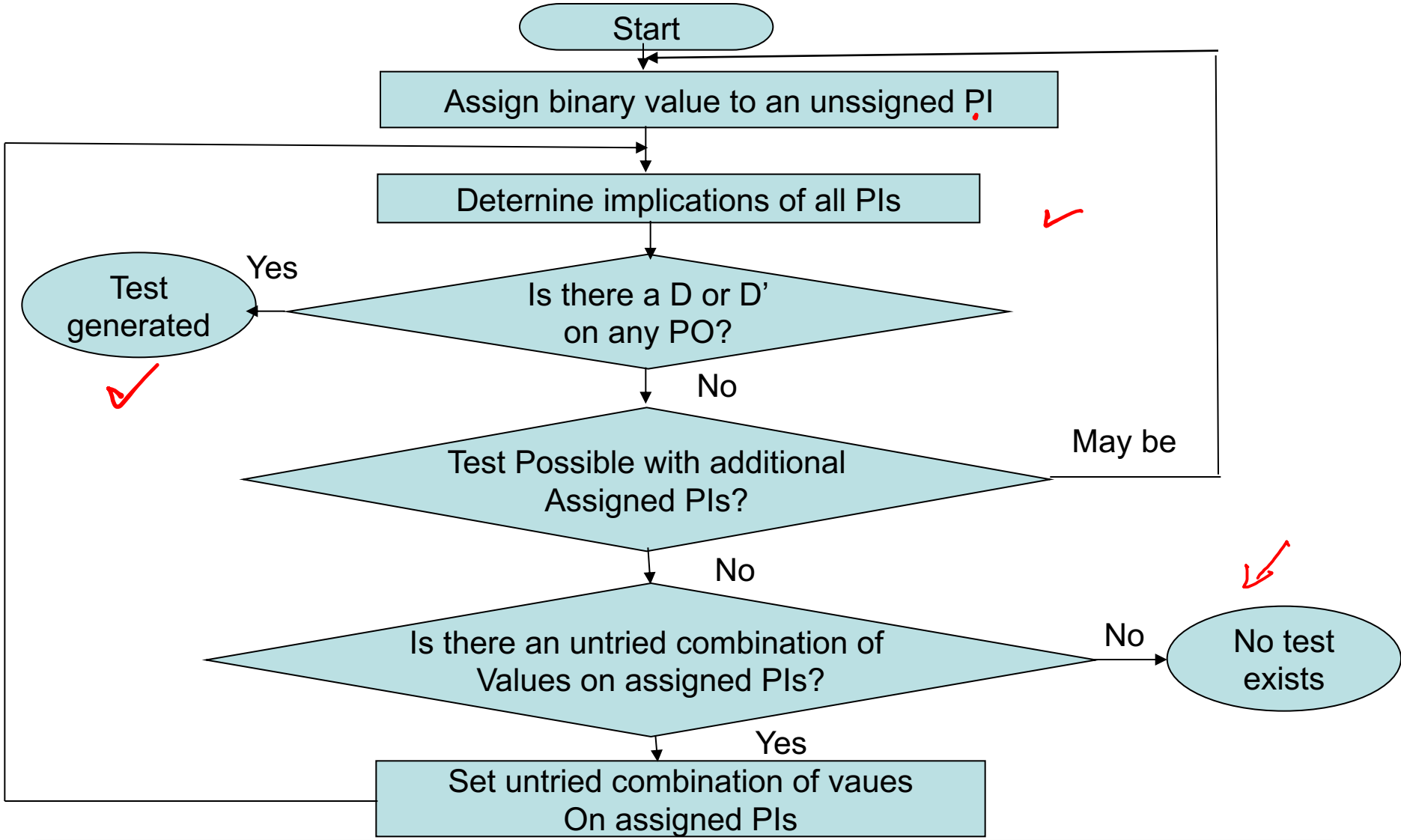


J – Frontier : A set of all gates whose output value is known but not implied by its input value

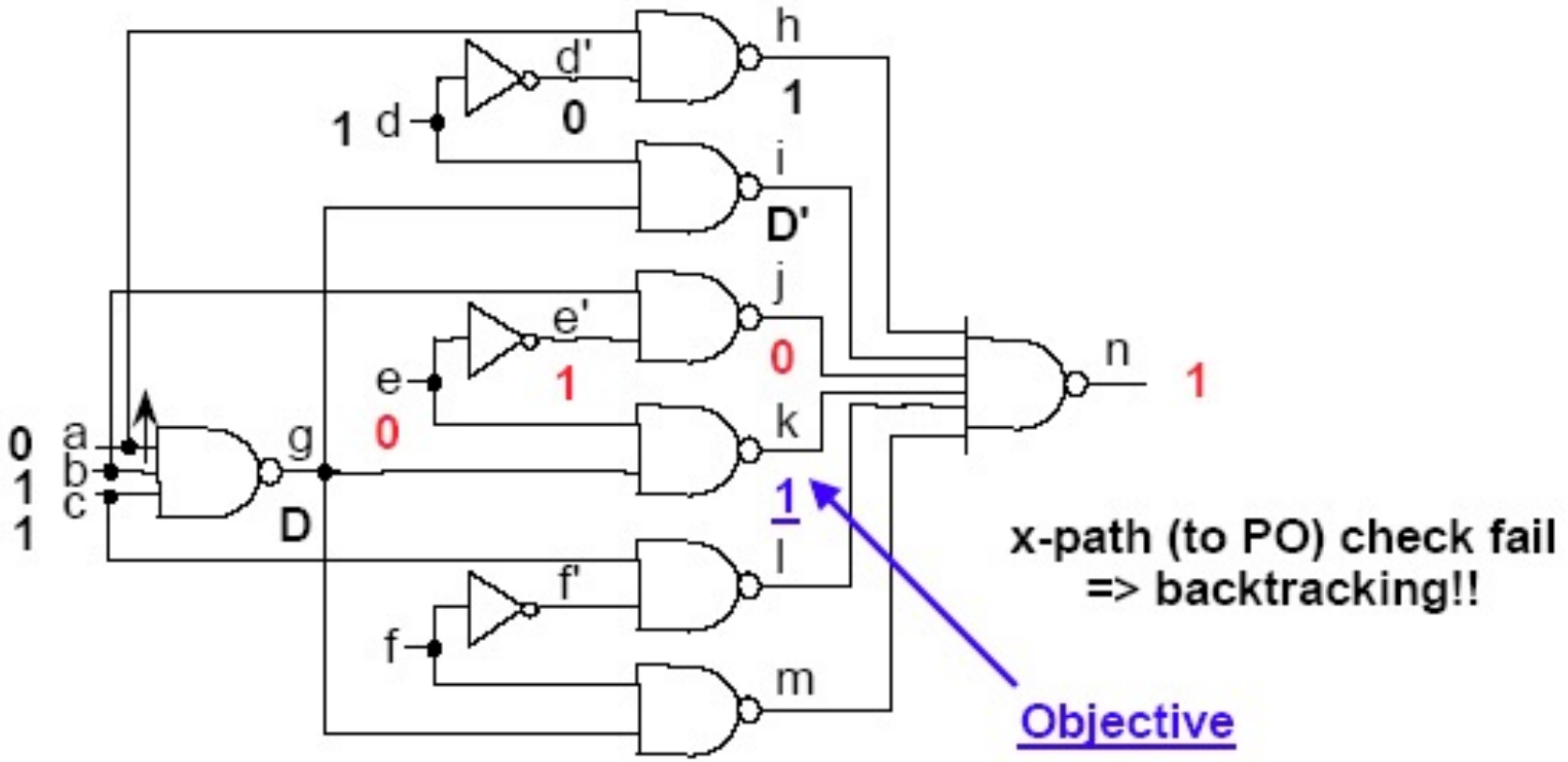
PODEM High-Level Flow

1. Assign binary value to unassigned PI
2. Determine implications of all PIs
3. Test Generated? If so, done.
4. Test possible with more assigned PIs? If maybe, go to Step 1
5. Is there untried combination of values on assigned PIs? If not, exit: untestable fault
6. Set untried combination of values on assigned PIs using objectives and backtrace. Then, go to Step 2

PODEM-Algorithm



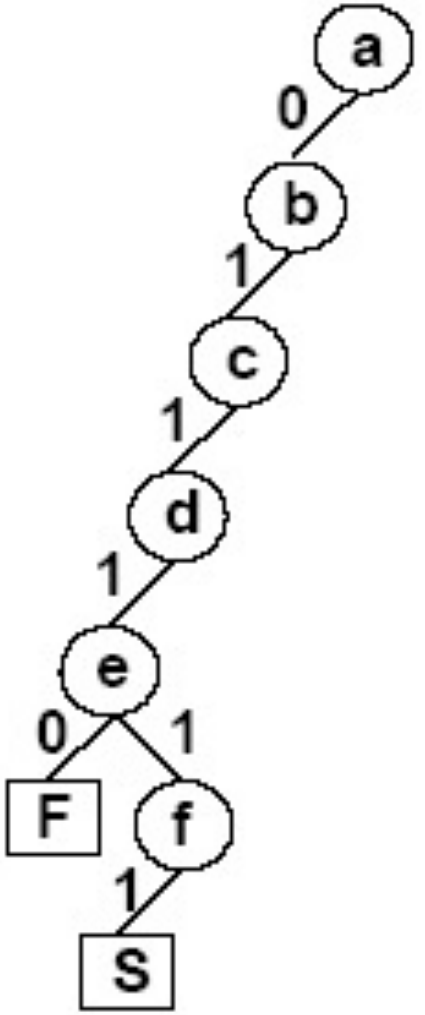
PODEM : Example



PODEM : Value Comp

Objective	PI assignment	Implications	D-frontier	Comments
a=0	a=0	h=1	g	
b=1	b=1		g	
c=1	c=1	g=D	i,k,m	
d=1	d=1	d?0 i=D	 k,m,n	
k=1	e=0	e?1 j=0 k=1 n=1	 m	 x-path check fail !!
	e=1	e?0 j=1 k=D	 m,n	reversal
l=1	f=1	f?0 l=1 m=D n=D		

PODEM : Decision Tree



Thank You