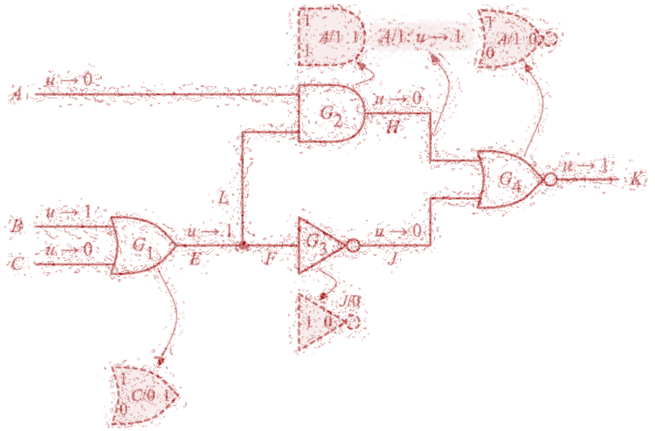
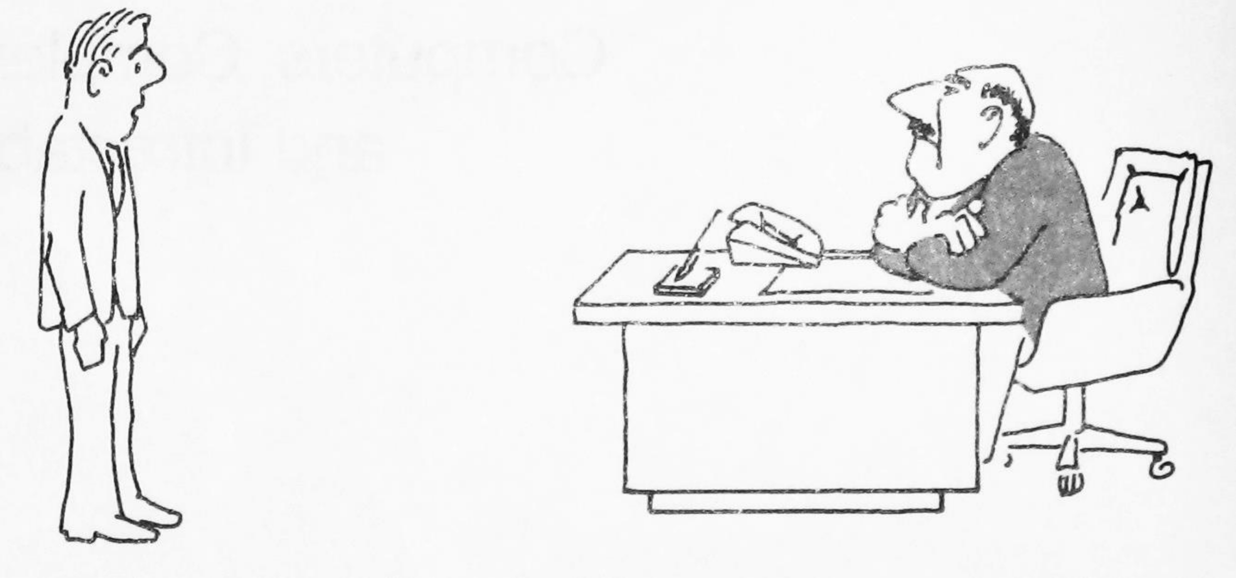
Fault Simulation

- Introduction
- Fault simulation techniques
- Comparison of fault simulation
- Alternatives to fault simulation
  - Toggle coverage
  - Fault sampling (1974)
  - Critical path tracing (1979)
- Issues of fault simulation
- Concluding remarks



Motivating Problem

- Boss: Why can’t we speed up design sign-off?
- You: But the fault simulation is slow…
- Boss: We do not need exact fault coverage. Just give me an estimation and DO IT FAST!



What Should You Do?

Alternatives to Fault Simulation

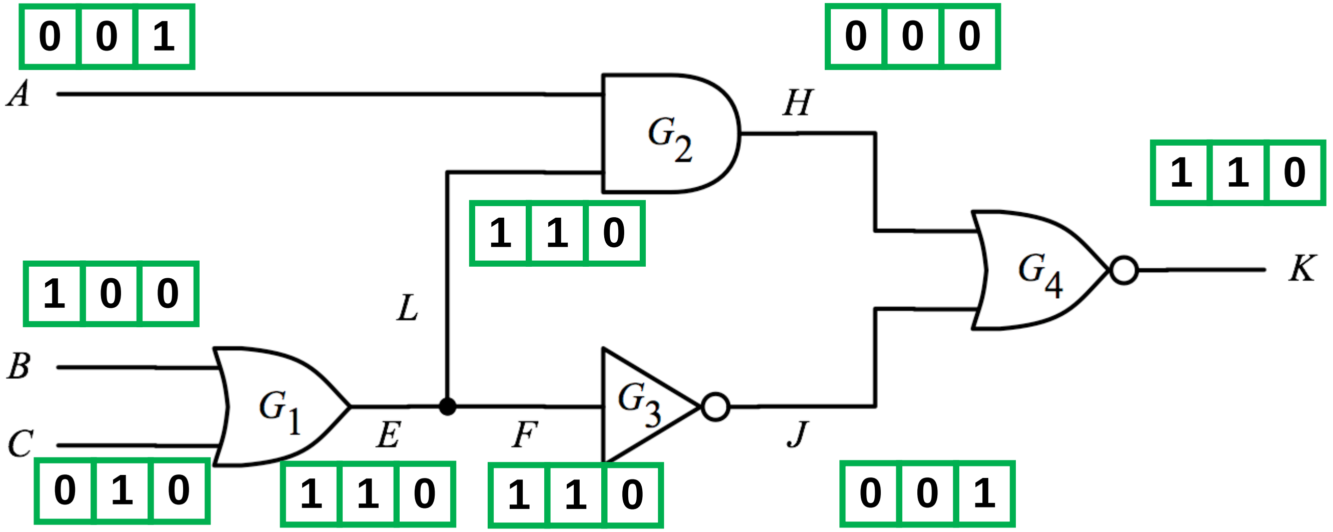
- **Q: Is exact fault simulation necessary?**
  - **Exact fault simulation is too expensive**
  - **Sometimes, we just want to know *approximate* fault coverage**
    - **with reasonable error**

- **Approximation is good for**
  - **DFT check in early design phase**
  - **Functional test pattern evaluation**
- **Approximation is NOT good for**
  - **ATPG**
  - **Diagnosis**

Toggle Coverage (DEF-1)

$$toggle\ coverage = \frac{\sum_{all\ nodes\ i} \#\ of\ different\ values\ in\ node\ i}{2 \times total\ \#\ of\ nodes}$$

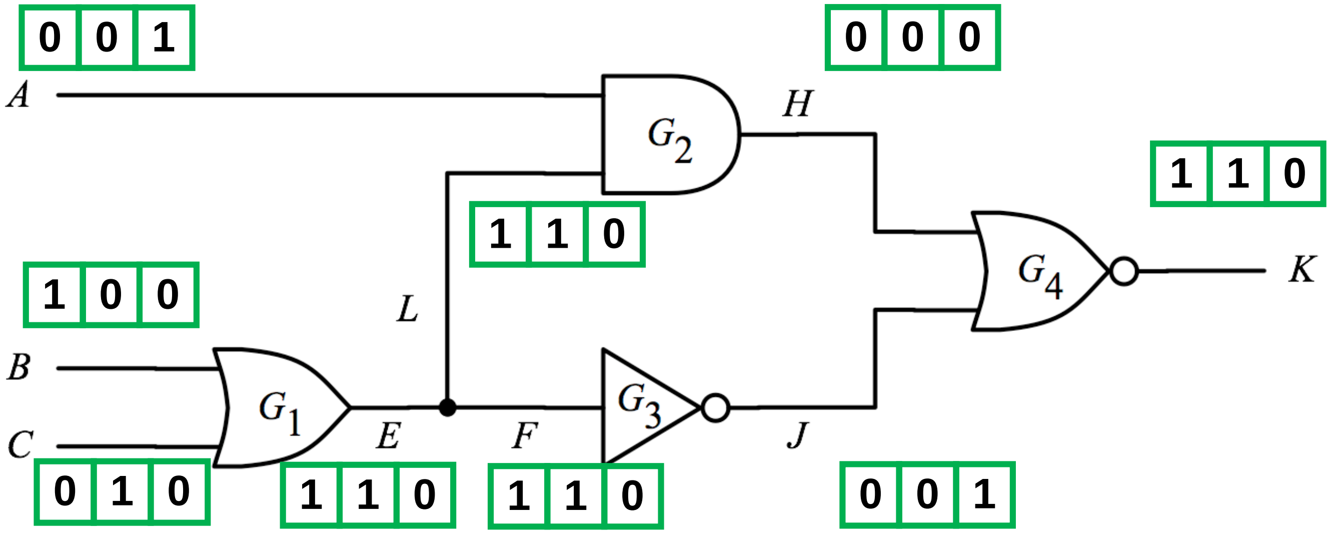
# = numbers



Toggle Coverage = 17/18 =94%

Toggle Coverage

- ☺ Advantage: Toggle coverage is easy to obtain
  - ♦ Logic simulation only, NO fault simulation
- ☹ Disadvantage: Toggle coverage is very optimistic
  - ♦ Fault activation only, NO fault propagation



TC is Upper Bound of FC

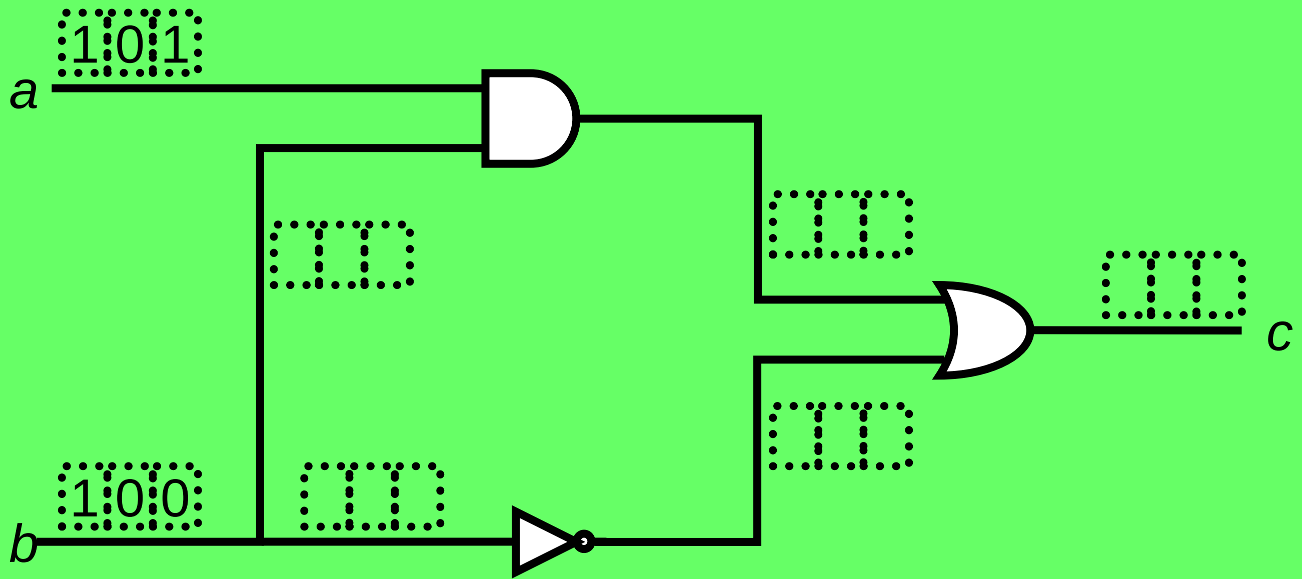
Fault Coverage = 11/18 = 61% see 5.3 P.10

Toggle Coverage = 17/18 =94%

Quiz

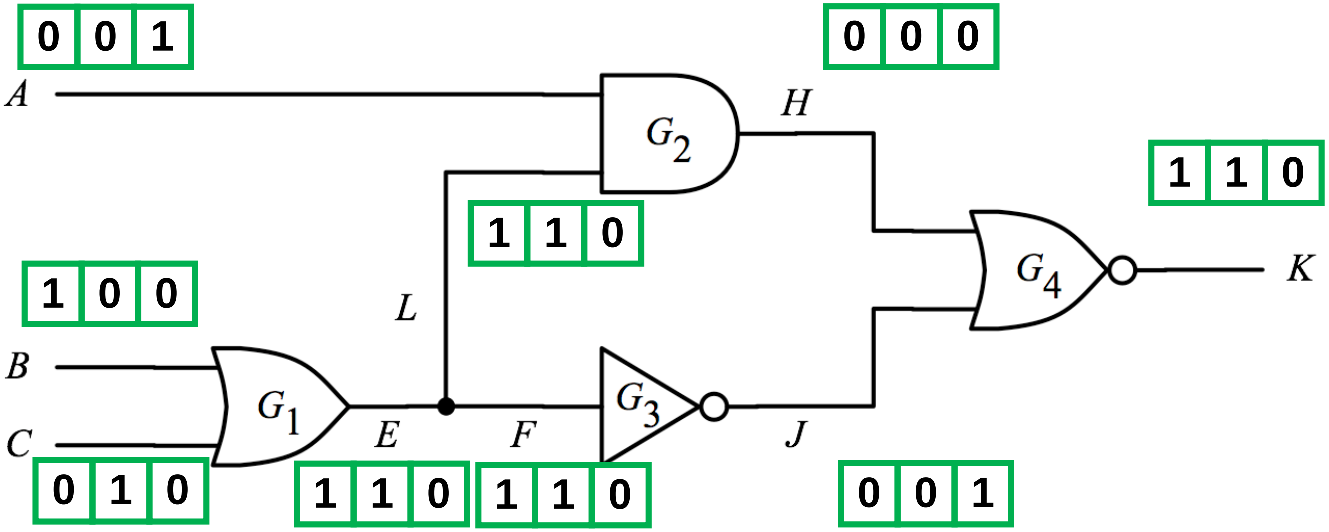
Q: Apply 3 patterns to this circuit of 7 nodes.   Toggle coverage =?

A:



Toggle Coverage (DEF-2)

$$\text{toggle coverage} = \frac{\sum_{\text{all nodes } i} \#\text{ of different } \underline{\text{transitions}} \text{ in node } i}{2 \times \text{total } \# \text{ of nodes}}$$

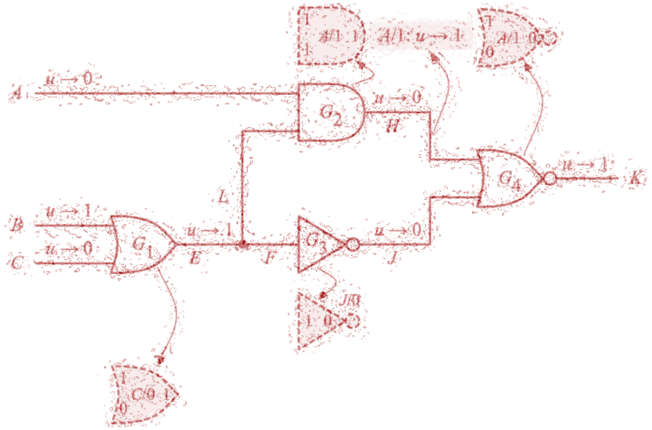


Toggle Coverage = 9/18 =50%

DEF-2 more stringent

Fault Simulation

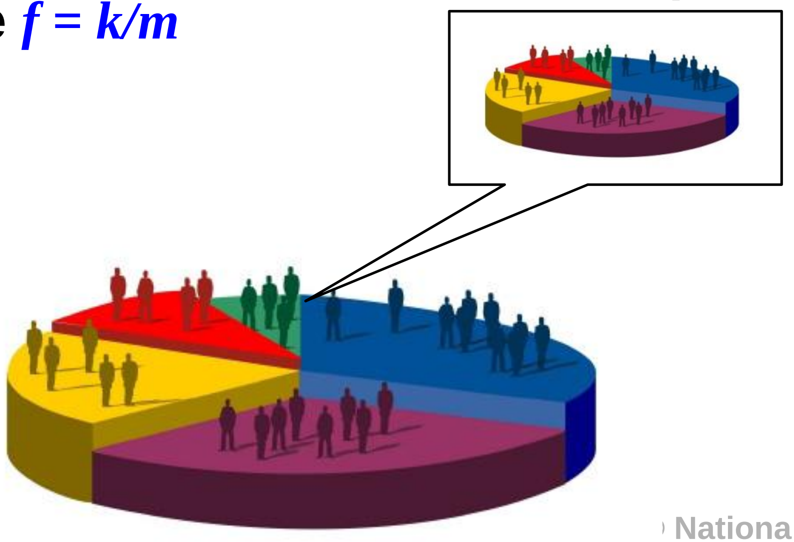
- Introduction
- Fault simulation techniques
- Comparison of fault simulation
- Alternatives to fault simulation
  - Toggle coverage
  - Fault sampling (1974)
  - Critical path tracing (1979)
- Issues of fault simulation
- Concluding remarks



Fault Sampling [Butler 74]

- Idea: Actual fault simulation of whole circuit is too slow
  - Can we just sample a small portion of faults?
  - Like *Polling* before election
- Notation
  - M: number of total faults
    m: number of sampled faults
    K: number of total faults detected (unknown)
    k: number of sampled faults detected
  - Actual fault coverage $F = K/M$ (unknown)
  - Sampled fault coverage $f = k/m$

sampled



$f=k/m$

$F=K/M$

total population

Fault Sampling (2)

- $P_k(M,m,K)$ = probability that a test detects k faults from a random sample size of m, given that it detects K faults from M total faults
  - P_k is *hypergeometric distribution* (discrete valued)

$$P_k(m,M,K) = \frac{C_k^K C_{m-k}^{M-K}}{C_m^M}$$

- For large M,
  - P_k can be approximated by normal distribution (continuous valued)
  - with mean μ_k and standard deviation σ_k

$$\mu_k = m\frac{K}{M} = mF$$

$$\sigma_k^2 = m\frac{K}{M}(1-\frac{K}{M})\frac{M-m}{M-1} \cong mF(1-F)(1-\frac{m}{M})$$

See papers for detailed derivation

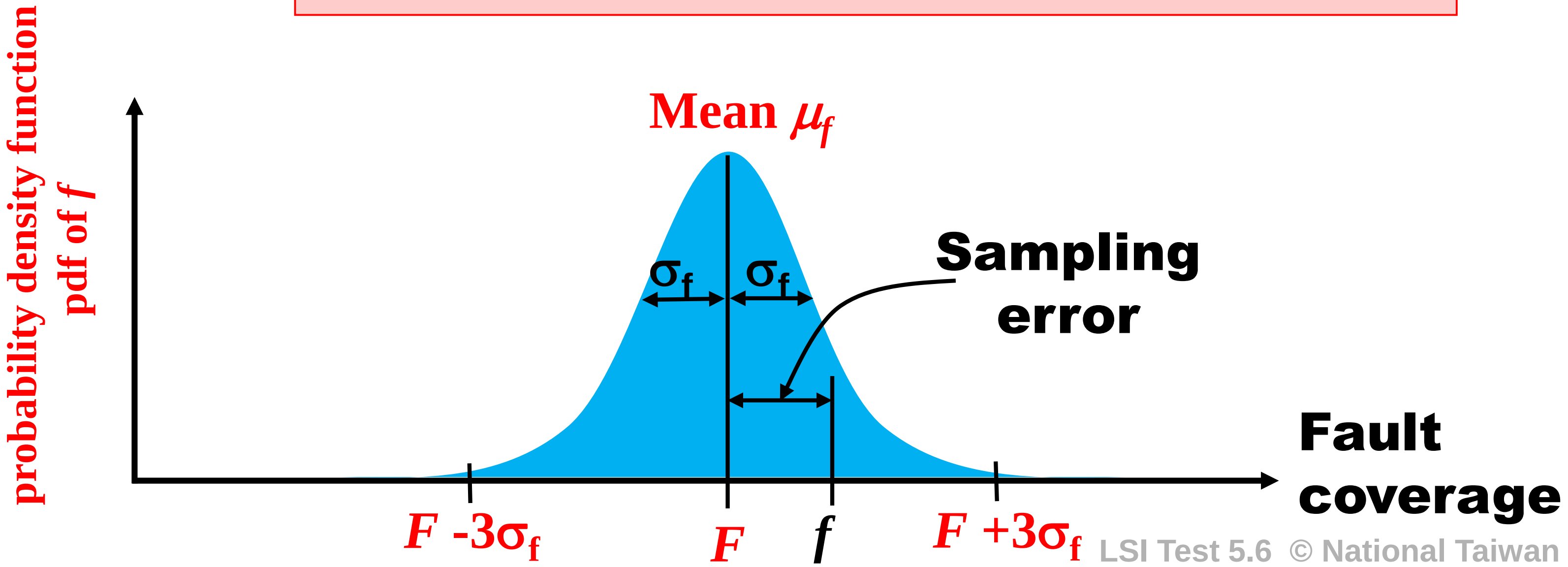
Fault Sampling (3)

- For large M, sampled fault coverage f is random variable with normal distribution:

$$\boxed{\mu_f = \mu_k / m = F}$$

$$\sigma_f^2 = \sigma_k^2 / m^2 = F(1-F)(1-m/M)/m$$

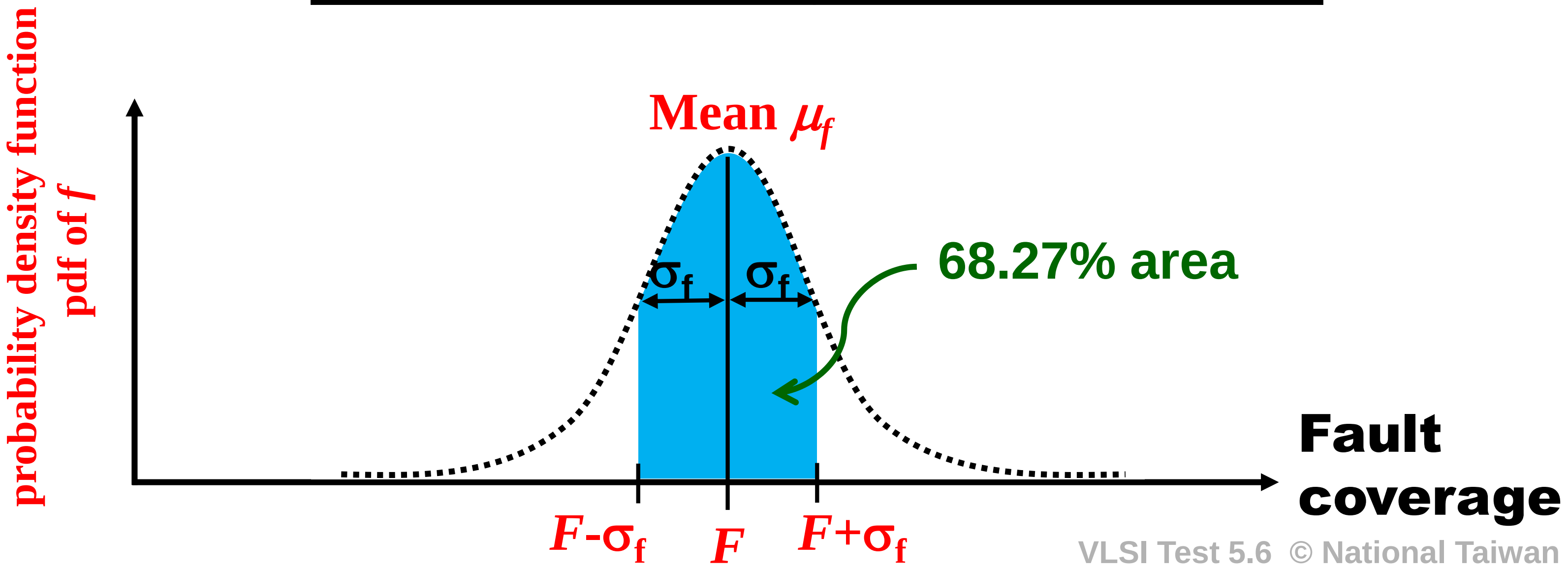
$$\boxed{\begin{aligned}\sigma_f &= \sqrt{F(1-F)(1-m/M)\ \ /m} \\ &\cong \sqrt{F(1-F)\ \ /m} \qquad\qquad \text{if } M >> m\end{aligned}}$$



Confidence Level

- ***Confidence Level*** **= probability that estimated fault coverage *f* falls in between the *confidence interval***
- **If we assume normal distribution:**

Confidence interval	Confidence level
$\mu_f \pm \sigma_f$	68.27%
$\mu_f \pm 2\sigma_f$	95.44%
$\mu_f \pm 3\sigma_f$	99.73%
$\mu_f \pm 4\sigma_f$	99.99%



Example

- M=1,000,000
- m=10,000 (1% of M)
- NOTE: F is unknown so we use μ_f

$$\sigma_f = \sqrt{F(1-F)(1-m/M)\ /m}$$
$$\cong \sqrt{F(1-F)\ /m}$$

μ_f (=F)	70%	80%	90%	95%
σ_f	0.5%	0.4%	0.3%	0.2%
68% CL	70%±0.5%	80%±0.4%	90%±0.3%	95%±0.2%
95% CL	70%±1%	80%±0.8%	90%±0.6%	95%±0.4%
99% CL	70%±1.5%	80%±1.2%	90%±0.9%	95%±0.6%

Accurate Prediction:
High CL and Small CI

Quiz

Q: Suppose FC=95% but we want to improve std σ_f to 0.1%.
How many samples do we need, m=?

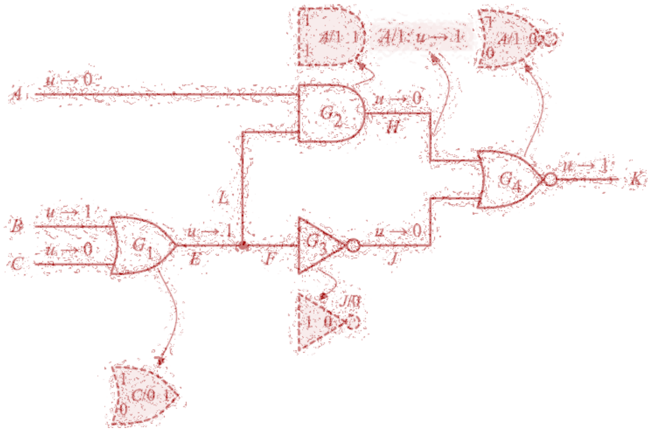
A:

$$\sigma_f = \sqrt{F(1-F)(1-m/M)\ /m}$$
$$\cong \sqrt{F(1-F)\ /m}$$

μ_f (=F)	70%	80%	90%	95%
σ_f	0.5%	0.4%	0.3%	0.2%
68% CI	70%±0.5%	80%±0.4%	90%±0.3%	95%±0.2%
95% CI	70%±1%	80%±0.8%	90%±0.6%	95%±0.4%
99% CI	70%±1.5%	80%±1.2%	90%±0.9%	95%±0.6%

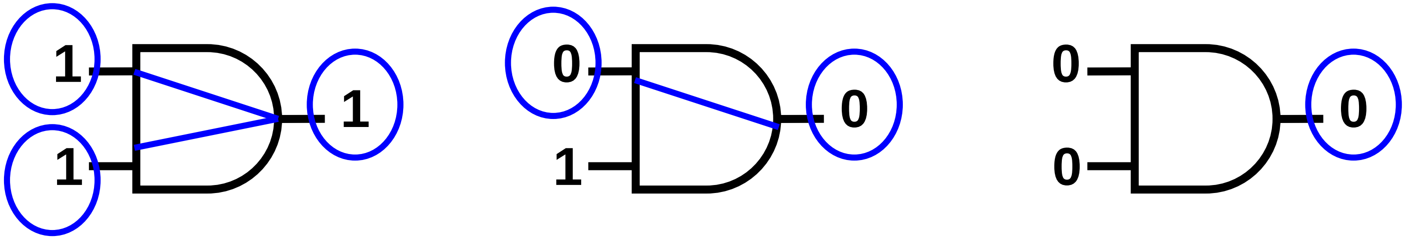
Fault Simulation

- Introduction
- Fault simulation techniques
- Comparison of fault simulation
- **Alternatives to fault simulation**
  - Toggle coverage
  - Fault Sampling (1974)
  - Critical path tracing (1979)
- Issues of fault simulation
- Concluding remarks



Critical Path Tracing (CPT) [Roth 79]

- *x is Critical Signal*
  - ♦ *x*'s value change causes some primary output values to change
- *Critical path*
  - ♦ All signals on this path that are critical signals
- Example: critical signals are circled

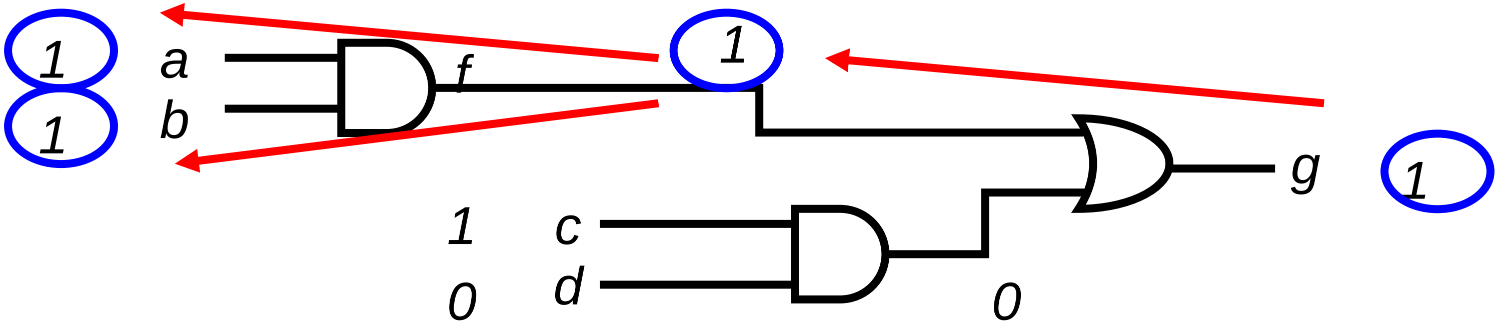


Note: In timing analysis, *critical path* has different definition

Critical path = Path with the longest path delay

CPT Algorithm [Abramovici 84]

- *Critical Path Tracing*: Start from primary outputs to primary inputs
  - If gate output is critical, backtrace its critical gate input(s)
- Example: fanout-free cone (FFC)
  - Critical signals: *a, b, f, g.* Critical paths: *afg* and *bfg*

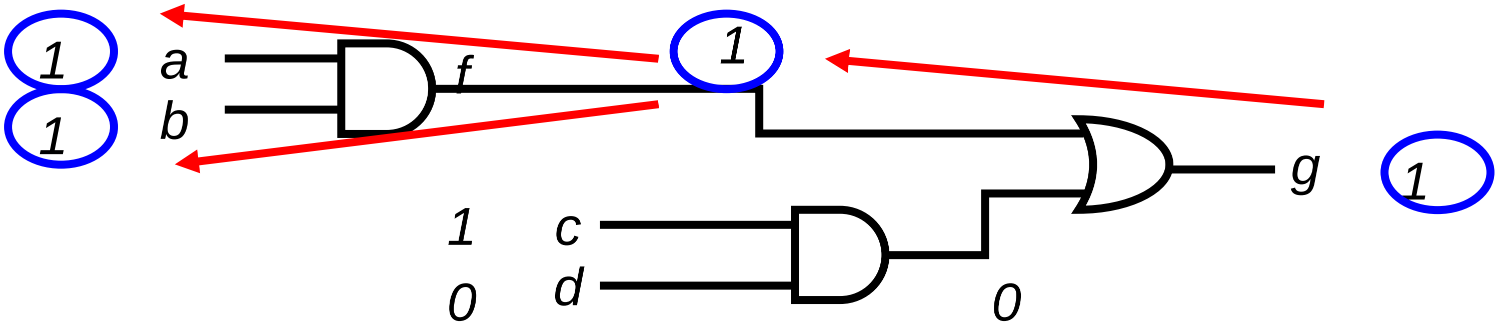


```
CPT (O)   /*O is a node*/
  1.  foreach gate input g of O
  2.      if (g is critical signal) then
  3.          CPT (g) ;
  4.      else return;
```

CPT Theorem

THEOREM:
If critical signal *g* has good value *v*, then *g* stuck-at *v'* fault is detected

- **Example: Detected faults: *a SA0, b SA0, f SA0, g SA0***

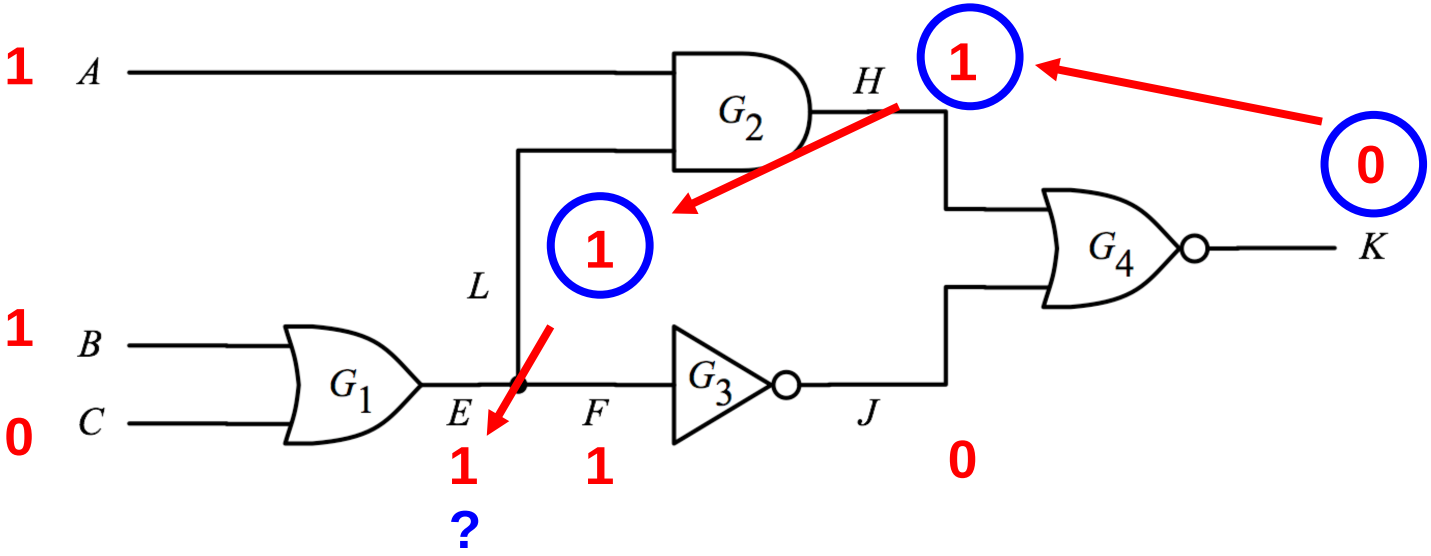


- **CPT Advantages:**
  - **CPT can be done in linear time**
  - **Only logic simulation needed for FFC**
  - **Used in first commercial ATPG tool, LASER**

CPT Is Very Fast … but

How about Fanout Stem?

- *L* is critical; *F* is non-critical
- *E* is non-critical



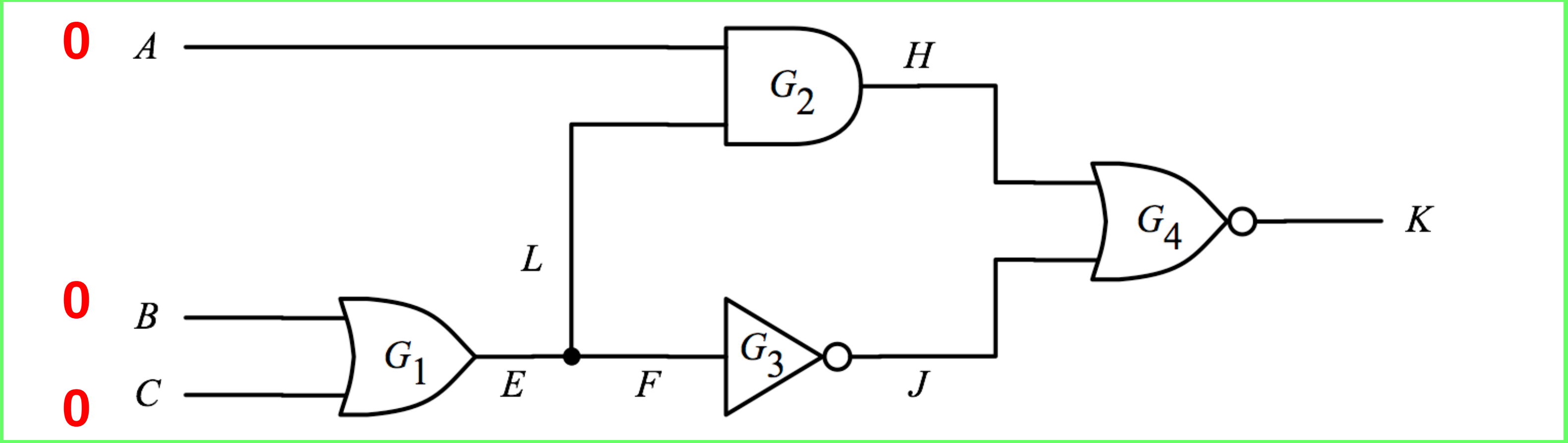
Is This Always The Case?

Fanout stems are non-critical?

Quiz

Q: Determine if fanout stem E is critical or not

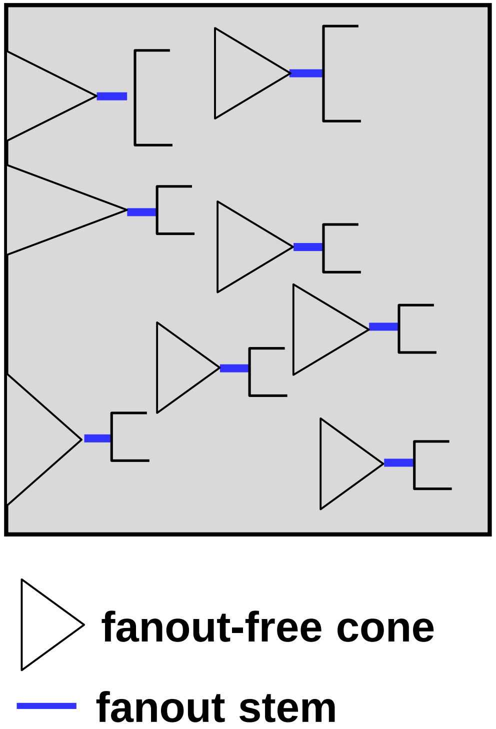
A:



CPT Does NOT Work Well when Fanout Reconverges

So… What Should We Do?

- 1. Count fanout stems as non-critical
  - Fast but fault coverage pessimistic
- 2. Only fault simulate fanout stems, CPT in fanout-free cones
  - Slow but fault coverage accurate



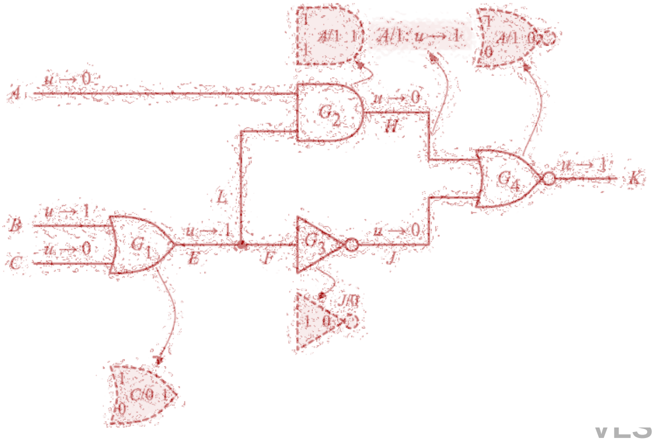
Summary

- **Alternatives to fault simulation, fast but inaccurate**

① **Toggle coverage**
- ♦ **Only logic simulation needed**

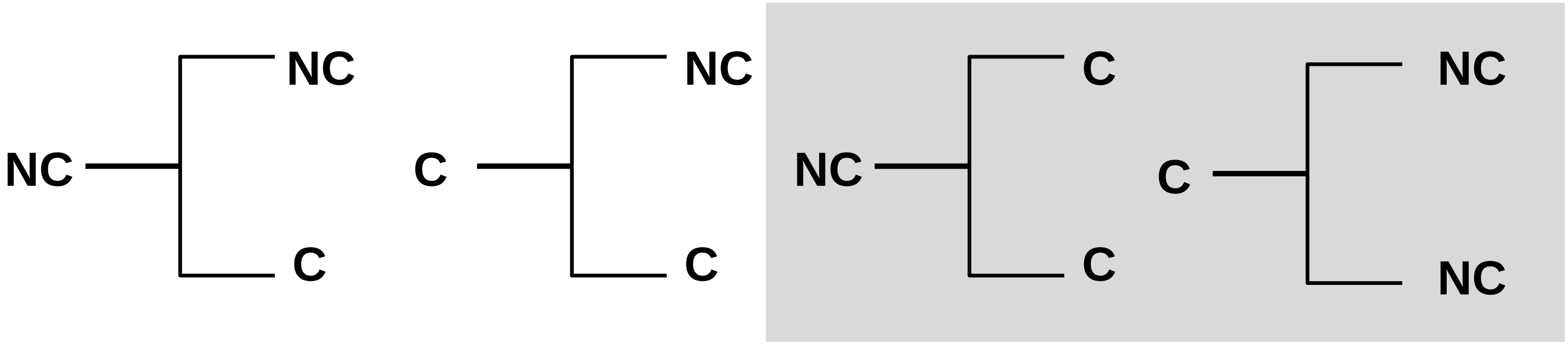
② **Fault Sampling**
- ♦ **Only fault simulate a small portion of faults**
- ♦ **Confidence interval and confidence level**

③ **Critical path tracing**
- ♦ **Linear time for fanout-free cones**
- ♦ **Still need fault simulation when fanouts reconverge**



FFT

- Q1: For CPT, all four cases are possible.
  - We have shown first two cases
  - Find example for latter two cases?



- Q2: Toggle coverage ≥ fault coverage. Can you prove it?
  - assume DEF-1 toggle coverage