

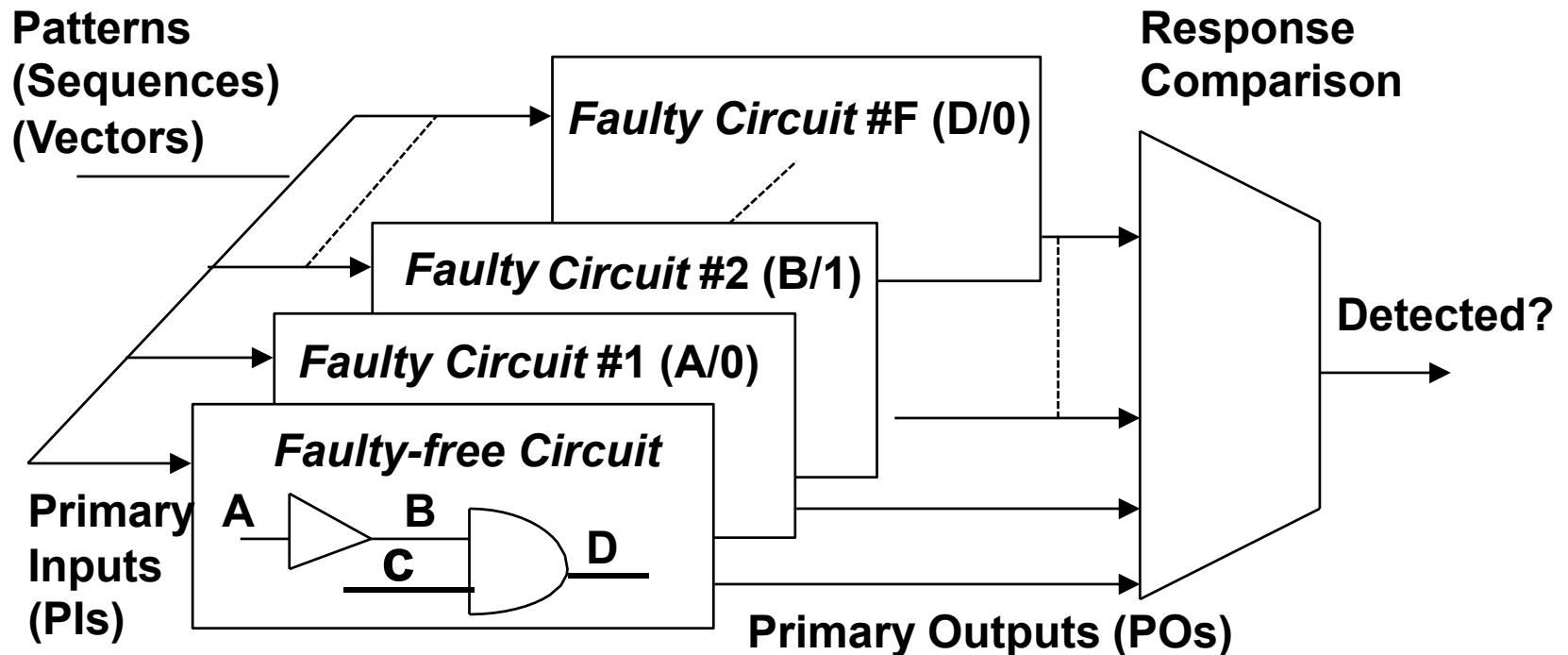
Fault Simulation

- **Introduction**
- **Classical Fault Simulation**
- **Modern Fault Simulation for Combinational Circuits**
- **Hardware Approaches to Fault Simulation**

Why Fault Simulation?

- 1. To evaluate the quality of a test set**
 - usually in terms of fault coverage**
- 2. To incorporate into ATPG for test generation**
 - due to its lower complexity**
- 3. To construct fault dictionary**
 - for post-test diagnosis**

Conceptual Fault Simulation

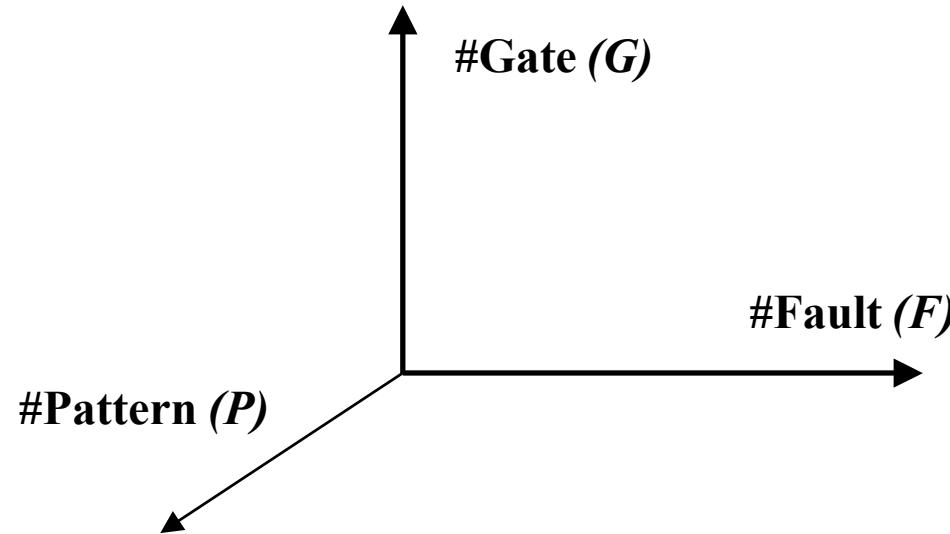


- Logic simulation on both good (fault-free) and faulty circuits

Some Basics for Logic Simulation

- For fault simulation purpose, mostly the gate delay is assumed to be zero unless the delay faults are considered. Our main concern is the functional faults.
- The logic values can be either two (0, 1) or three values (0, 1, X). For delay fault, more values will be needed.

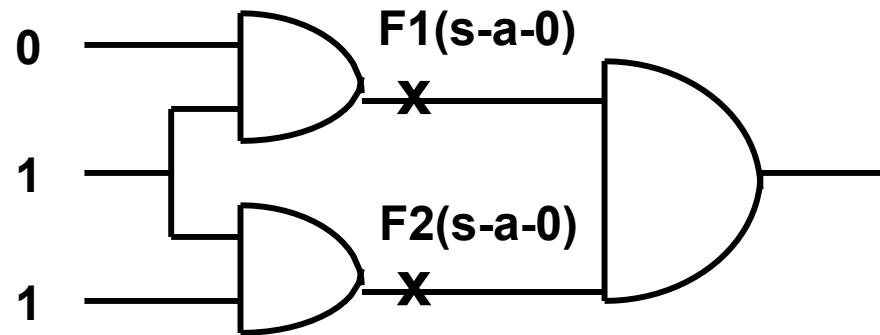
Complexity of Fault Simulation



- Complexity = $P * F * G \sim O(G^3)$ with single s-a faults
- The complexity is higher than logic simulation, $O(G^2)$, but is much lower than test pattern generation.
- In reality, the complexity is much lower due to fault dropping and advanced techniques.

Characteristics of Fault Simulation

- Fault activity with respect to fault-free circuit is often sparse both in time and in space.
- For example, F1 is not activated by the given pattern, while F2 affects only the lower part of this circuit.



- The efficiency of a fault simulator depends on its ability to exploit these characteristics.

Classical Fault Simulation Techniques

- **Common Characteristics:**
 - In general, no restriction on the circuit types.
 - Developed before VLSI era.
- **Serial Fault Simulation**
 - trivial single-fault single-pattern
- **Parallel Fault Simulation**
- **Deductive Fault Simulation**
- **Concurrent Fault Simulation**

Parallel Fault Simulation

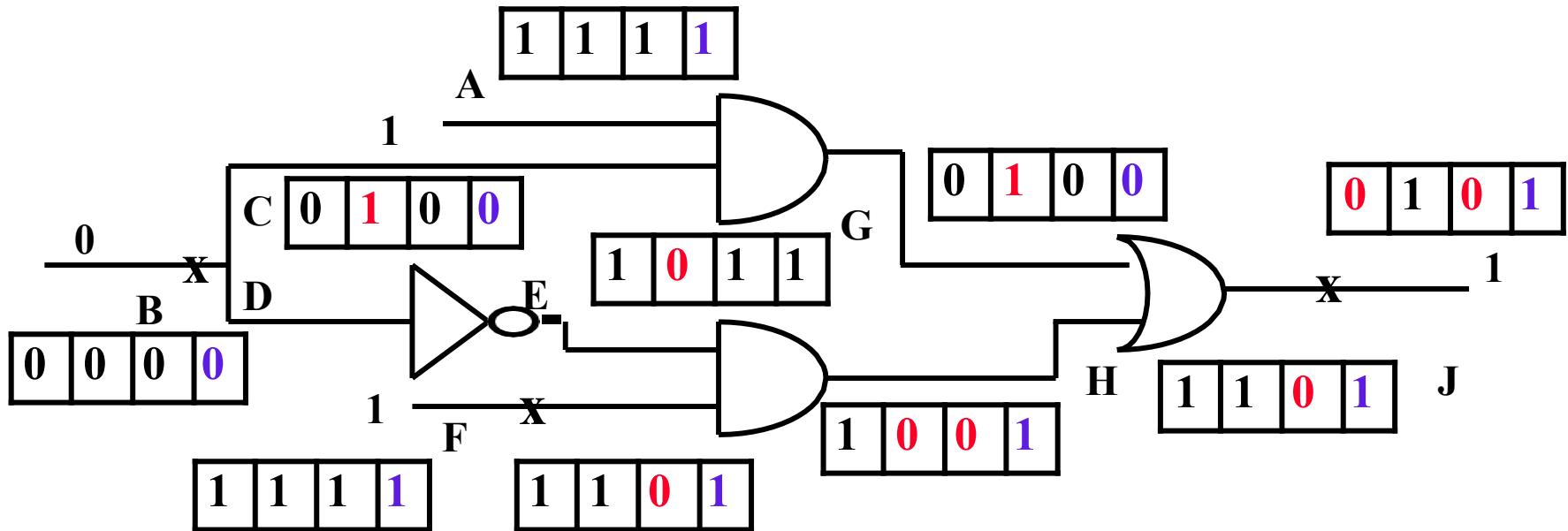
- Taking advantage of inherent parallel operation of computer words to simulate faulty circuits in parallel with fault-free circuit
 - the number of faulty circuits, or faults, can be processed parallelly is limited by the word length.
- Straightforward and memory efficient
- Some weaknesses:
 - An event, a value change, of a single fault or fault-free circuit leads to the computation of the entire word.
 - The fault-free logic simulation is repeated for the number of passes.

Example of Parallel Fault Simulation

- Consider three faults: B/1, F/0, and J/0
 - Bit-space:

| | | | |
|-----|-----|-----|----|
| J/0 | B/1 | F/0 | FF |
|-----|-----|-----|----|

 where FF = Fault-free



Deductive Fault Simulation

- **Each line contains a list of faults that have fault effects on this line.**
 - Only the faults with fault effects, or difference w.r.t. fault-free circuit, is retained in the list.
- **The propagation of such lists can be based on set operations considering the gate types and values.**
 - The list update is performed with each new pattern, which is not efficient in computer.
 - The list may dynamically grow in size, which incurs memory explosion problem.

Fault List Propagation Rule

- Let the gate $G = F(A, B)$ with input lists LA and LB and output list LG to be updated.
- If in the fault-free circuit,
 - the value of G is 0(1), use $F(F^-)$
 - the value of gate input A is 0(1), replace A in logic expression by $LA(LA^-)$ and A^- by $LA-(LA)$.
- Replace $*$ by *intersection* and $+$ by *union*.
- Add $G/1(G/0)$ to the list LG .

Fault List Propagation

- Logic deduction instead of circuit simulation

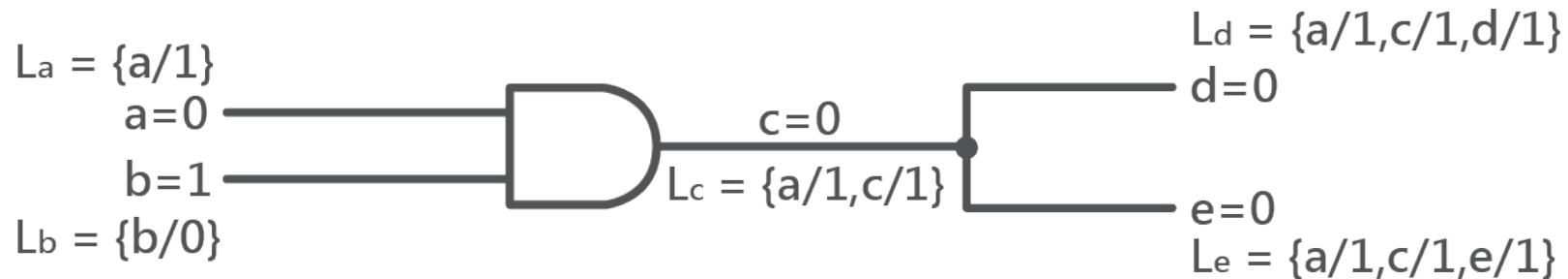
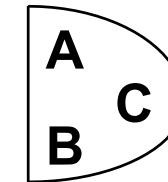


Illustration of Fault List Propagation

Consider a two-input AND gate



Case 1: $A=1, B=1, C=1$ at fault-free,

$$L_C = L_A + L_B + \{C/0\}$$

Case 2: $A=1, B=0, C=0$ at fault-free,

$$L_C = L_A^- * L_B + \{C/1\}$$

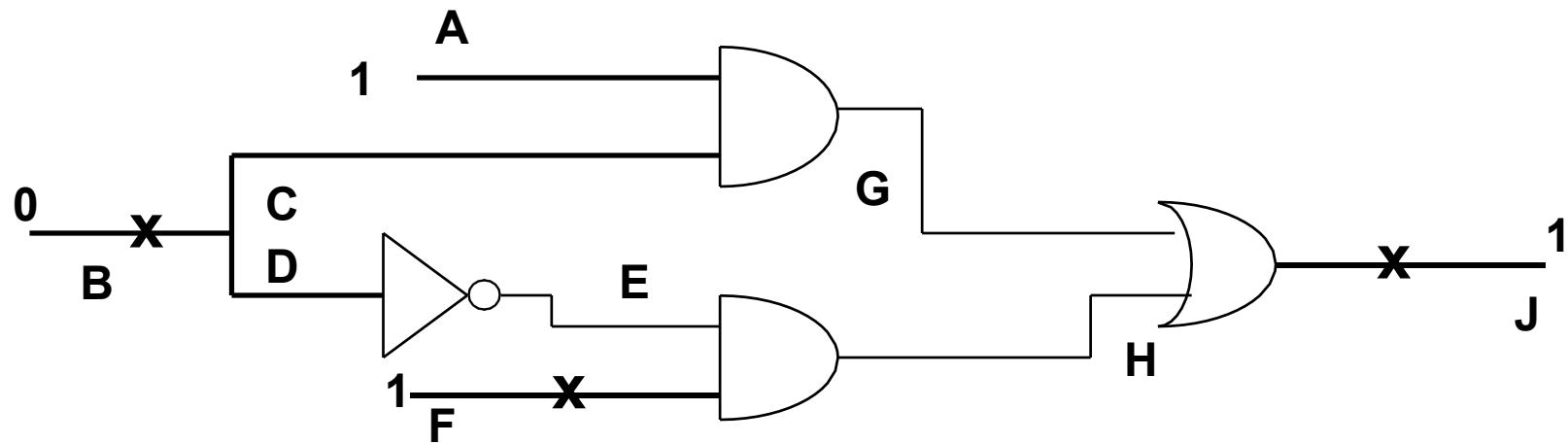
Case 3: $A=0, B=0, C=0$ at fault-free,

$$L_C = L_A * L_B + \{C/1\}$$

L_A^- is the set of all fault not in L_A

Example of Deductive Simulation Ia

- Consider 3 faults: B/1, F/0, and J/0

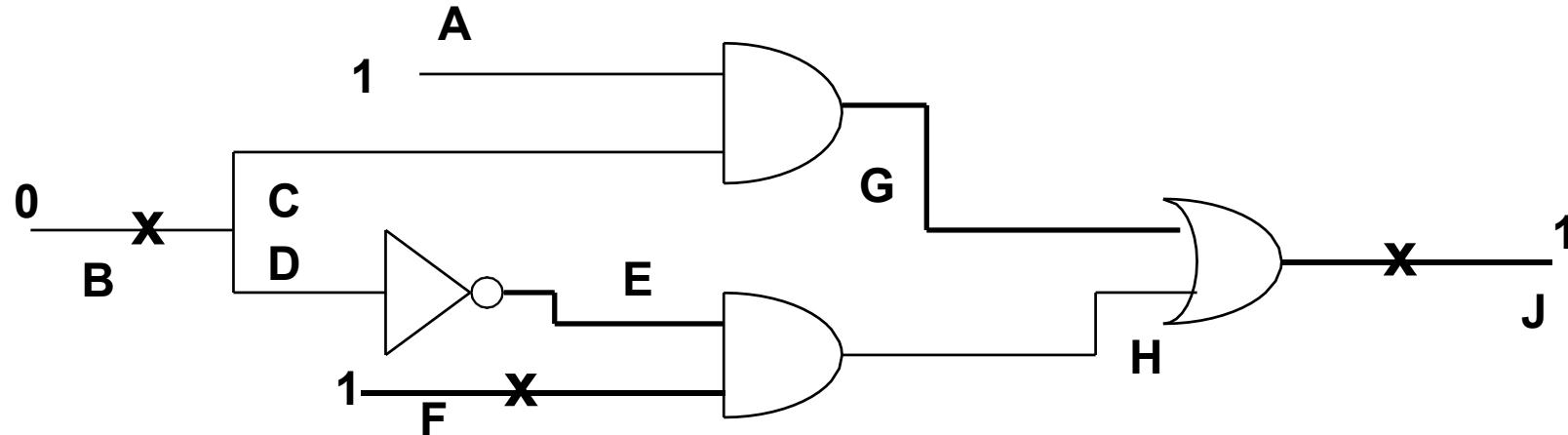


$$LB = \{B/1\}, \quad LF = \{F/0\}, \quad LA = 0$$

$$LC = LD = \{B/1\}$$

Example of Deductive Simulation Ib

- Consider 3 faults: B/1, F/0, and J/0

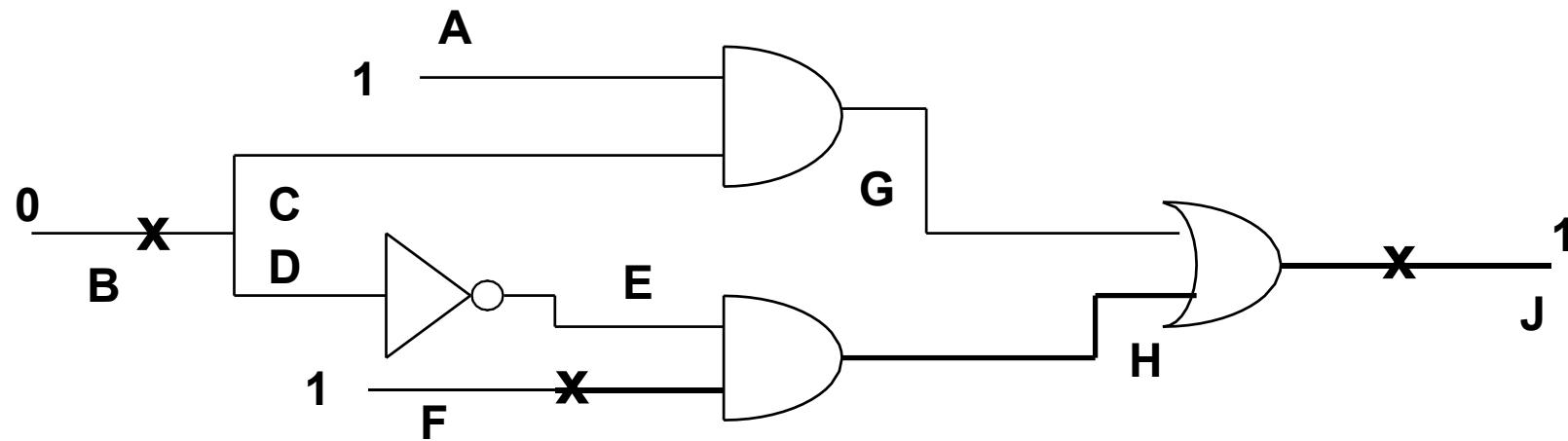


$LB = \{B/1\}$, $LF = \{F/0\}$, $LA = 0$, $LC = LD = \{B/1\}$

$LG = (LA - *LC) = \{B/1\}$, $LE = LD = \{B/1\}$

Example of Deductive Simulation Ic

- Consider 3 faults: B/1, F/0, and J/0



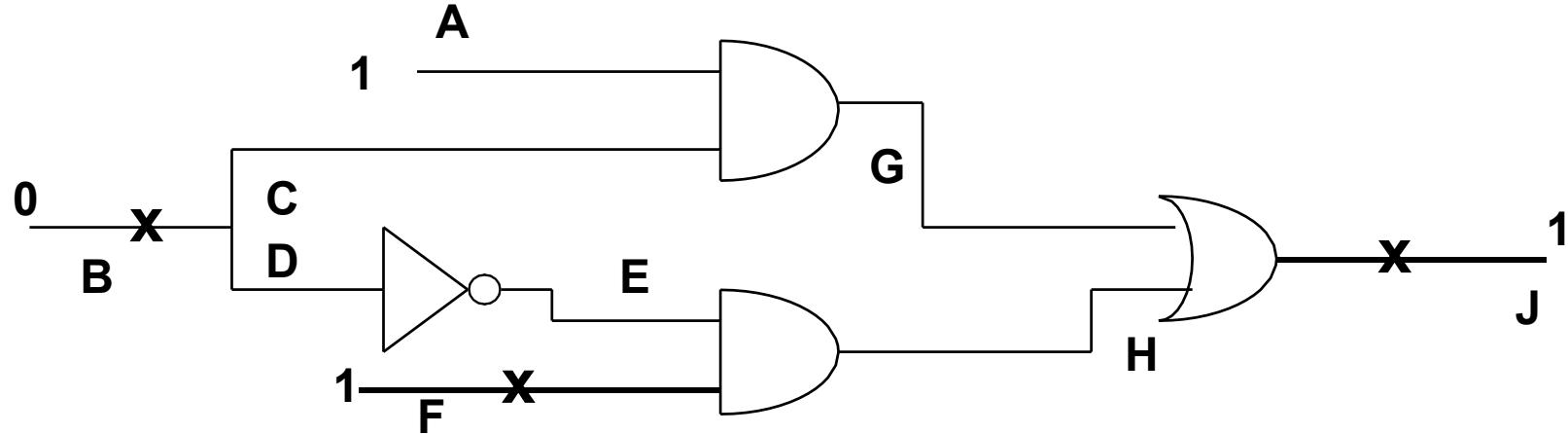
$$LB = \{B/1\}, LF = \{F/0\},$$

$$LC = LD = \{B/1\}, LG = \{B/1\},$$

$$LE = \{B/1\}, LH = (LE + LF) = \{B/1, F/0\}$$

Example of Deductive Simulation Id

- Consider 3 faults: B/1, F/0, and J/0



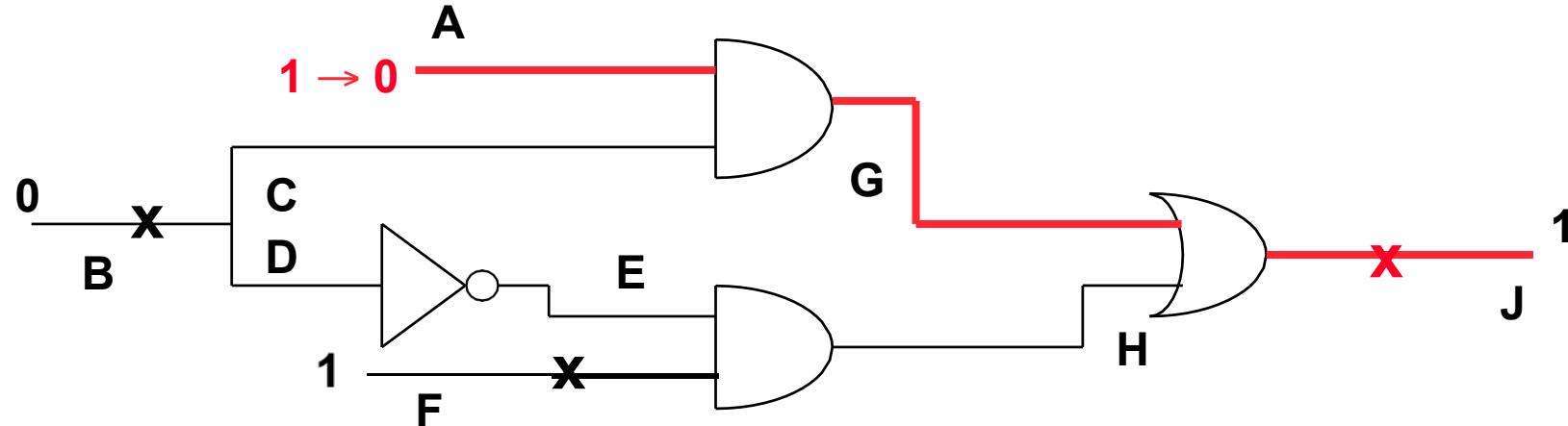
$L_B = \{B/1\}$, $L_F = \{F/0\}$,

$L_C = L_D = \{B/1\}$, $L_G = \{B/1\}$,

$L_E = \{B/1\}$, $L_H = \{B/1, F/0\}$, $L_J = \{F/0, J/0\}$

Example of Deductive Simulation II

- When A changes from 1 to 0



$$LB = \{B/1\}, \quad LF = \{F/0\},$$

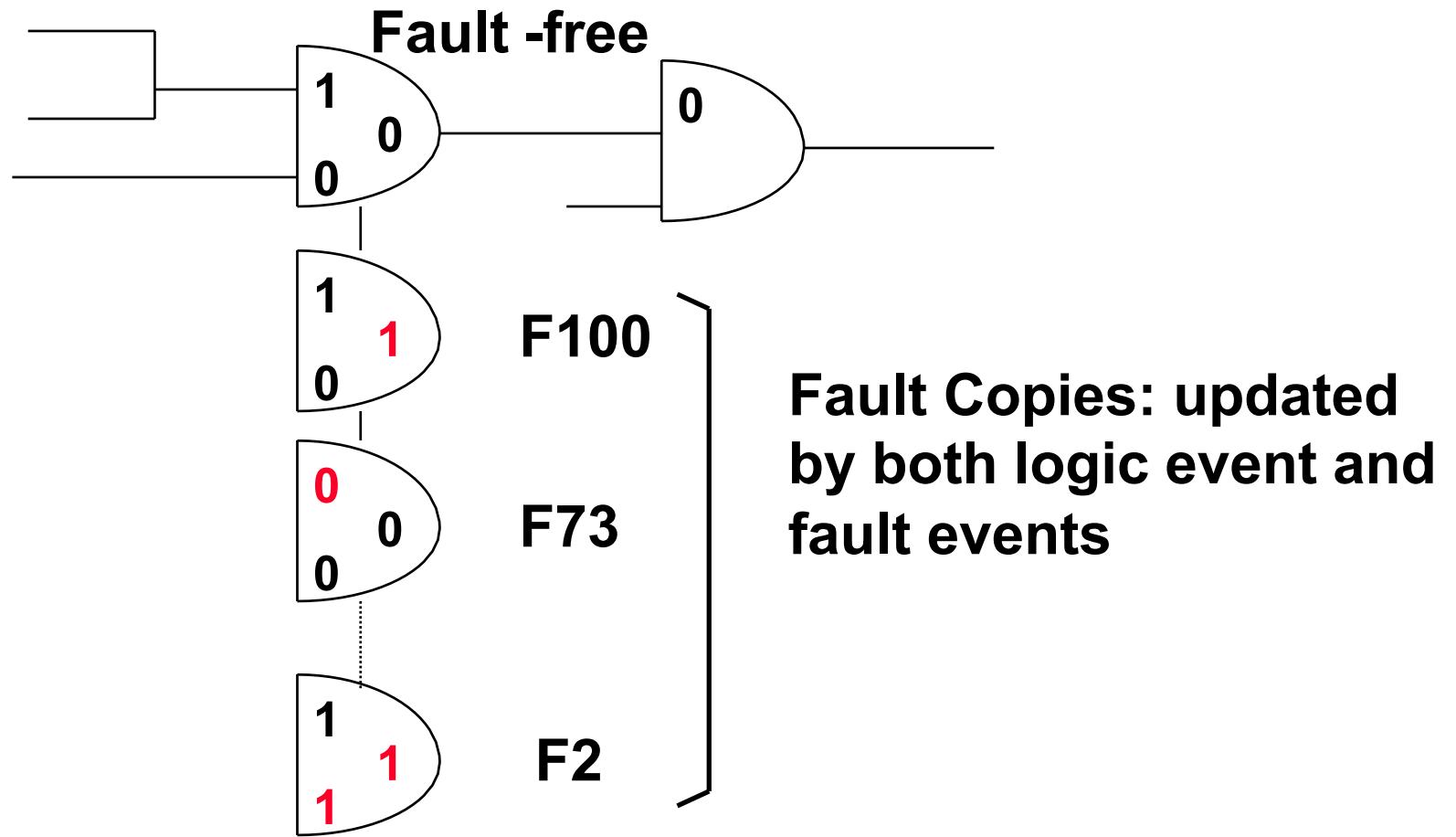
$$LC = LD = \{B/1\}, \quad LG = 0,$$

$$LE = \{B/1\}, \quad LH = \{B/1, F/0\}, \quad LJ = (LG \cdot LH) = \{B/1, F/0, J/0\}$$

Concurrent Fault Simulation

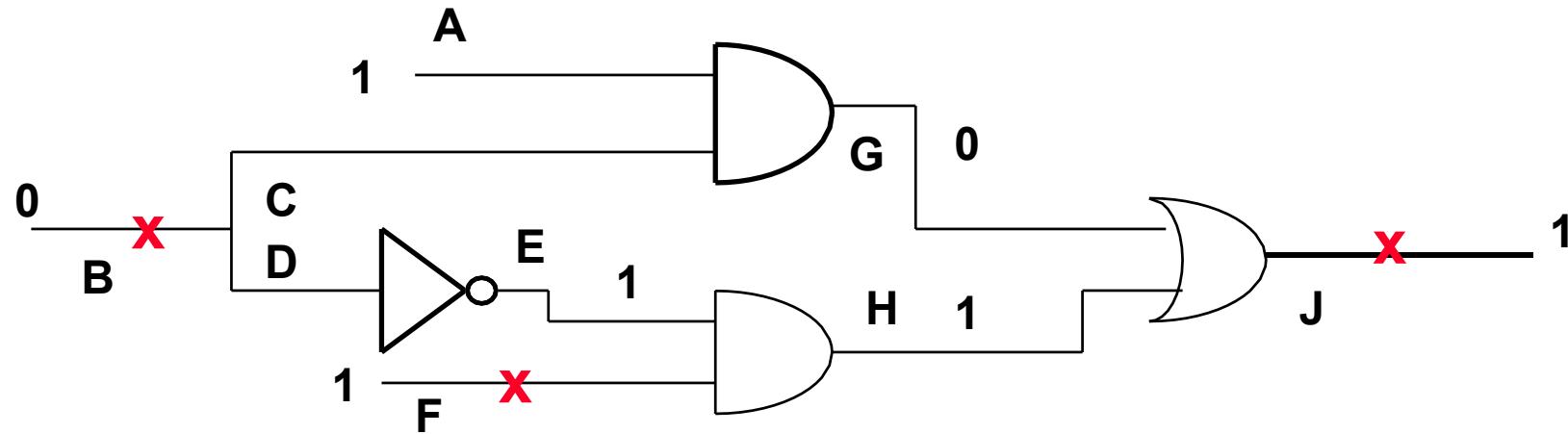
- **Each gate retains a list of fault copies each of which stores the status of a fault exhibiting difference from fault-free values.**
- **Simulation mechanism is similar to the conceptual fault simulation except that only the dynamical difference w.r.t. fault-free circuit is retained.**
- **Very versatile in circuit types and gate delays**
- **Although theoretically all faults in a circuit can be processed in one pass, memory explosion problem restricts the fault number in each pass.**

Concurrent Fault Simulation



Example of Concurrent Simulation I

- Consider 3 faults: B/1, F/0, and J/0



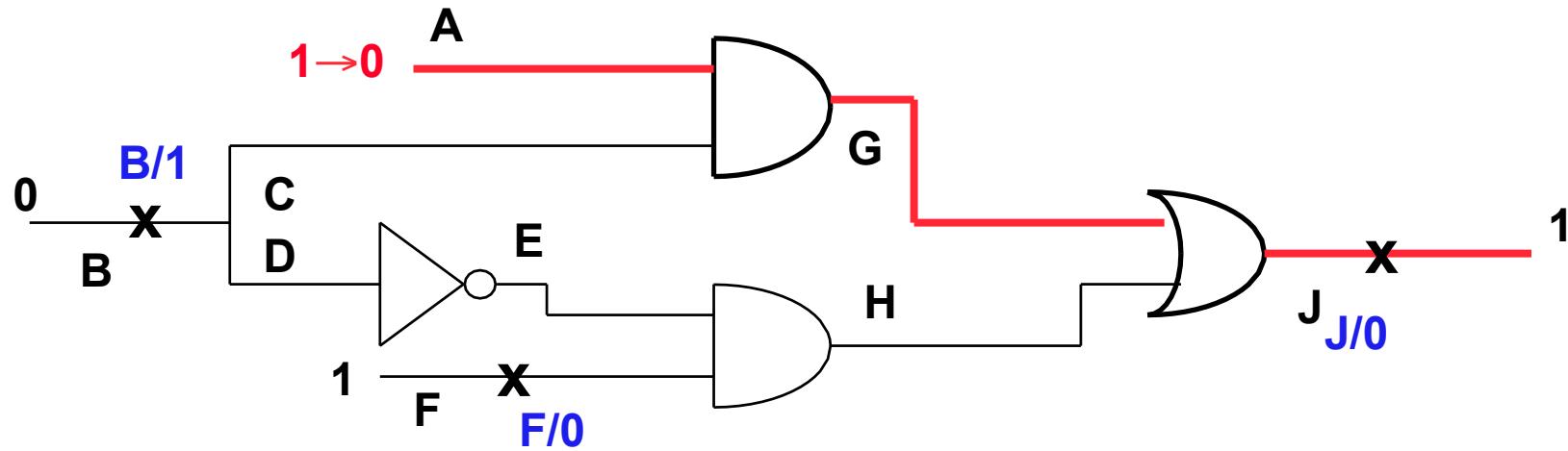
$$LG = \{10_0, B/1:11_1\} \quad LE = \{0_1, B/1:1_0\}$$

$$LH = \{11_1, B/1:01_0, F/0:10_0\}$$

$$LJ = \{01_1, B/1:10_1, F/0:00_0, J/0:01_0\}$$

Example of Concurrent Simulation II

- When A changes from 1 to 0



$$LG = \{00_0, B/1:01_0\} \quad LE = \{0_1, B/1:1_0\}$$

$$LH = \{11_1, B/1:01_0, F/0:10_0\}$$

$$LJ = \{01_1, B/1:00_0, F/0:00_0, J/0:01_0\}$$

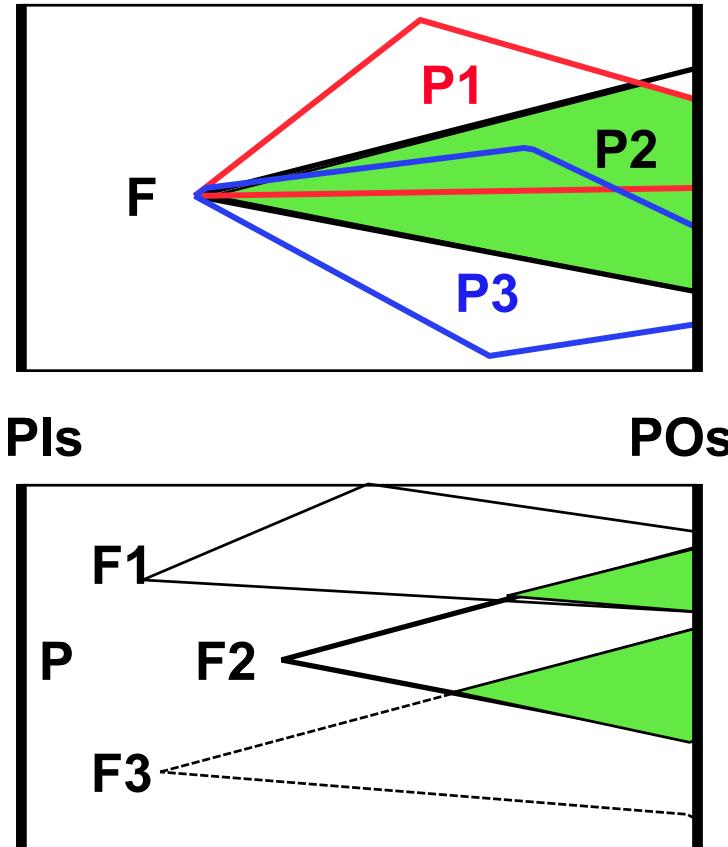
Modern Combinational Simulation Techniques

- Parallel pattern
- Other sophisticated techniques

Parallel-Pattern Single-Fault Propagation (PPSFP)

- Many patterns are simulated in parallel for both fault-free and faulty circuits.
- Coincident logic events of fault-free circuit from these patterns can be simulated in parallel.
 - reduction of logic event simulation time
- Coincident fault events of faulty circuits from these patterns can also be simulated in parallel.
 - reduction of fault event simulation time
- Simple and extremely efficient
 - basis of all modern combinational fault simulators

Comparison with Parallel-Fault



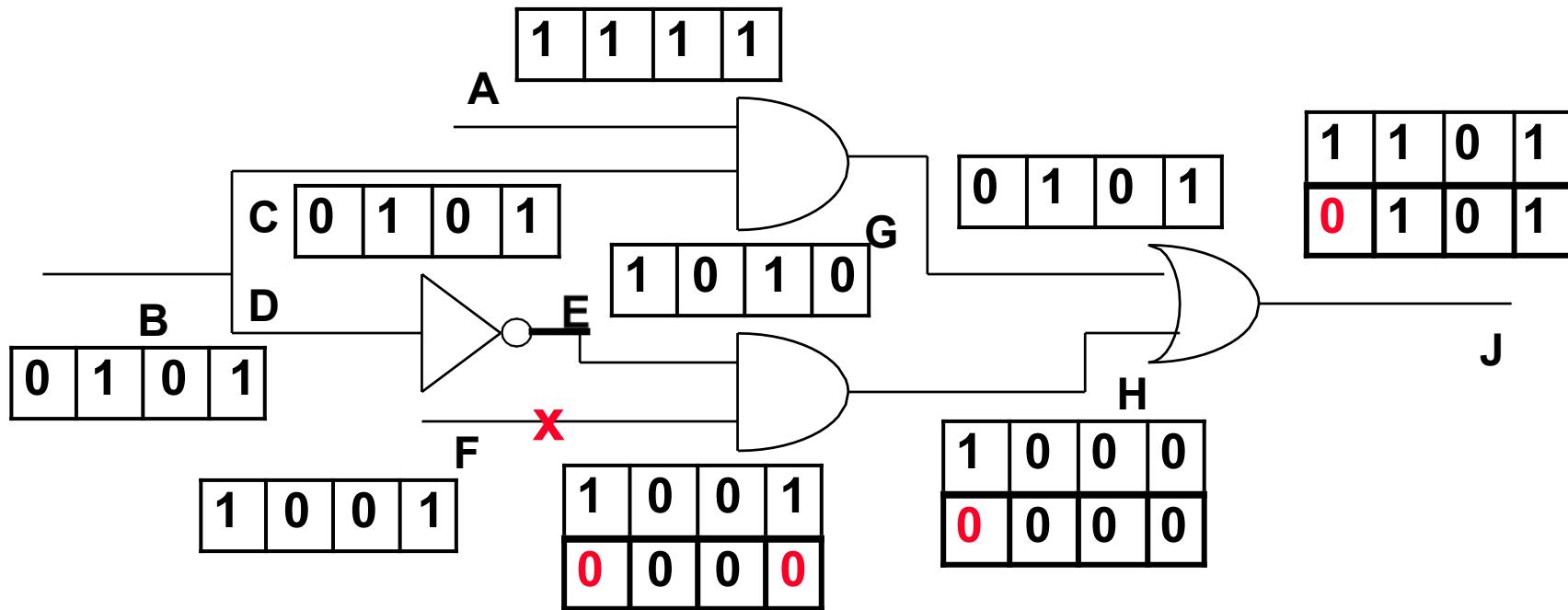
- Fault event maps for parallel-pattern (upper) and parallel-fault (lower)
- **Parallel-Pattern case:**
Consider three **patterns**: P1, P2, P3.
- **Parallel-Fault case:**
Consider three **faults**: F1, F2, F3.
- The **overlapping** of fault events is inherently much higher in parallel-pattern simulation, and hence more event can be done at the same time.

Example of Parallel Pattern Simulation

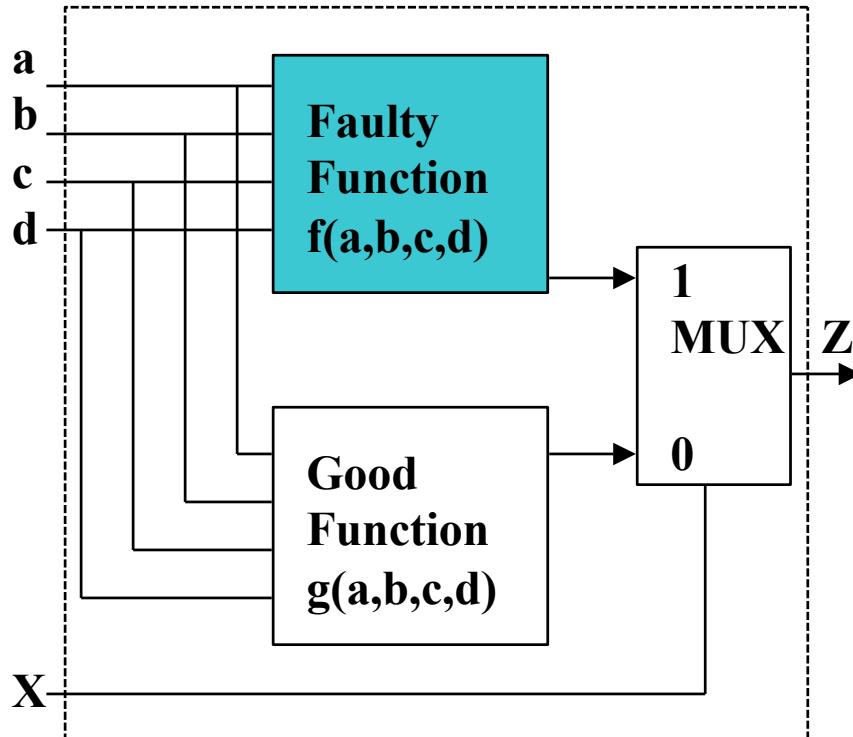
- Consider one fault **F/0** and four patterns: **P3,P2,P1,P0**
 - Bit-space:

| | | | |
|----|----|----|----|
| P3 | P2 | P1 | P0 |
|----|----|----|----|

 faulty value in **red**



Hardware Approache to Fault Simulation (FPGA)



A CLB with a dynamic fault injected (activated with $x=1$)

- **FPGA is a reconfigurable gate array which can be mapped from any logic circuits and emulated much faster than software simulation.**
- **The fault insertion process is slow. One way to minimize the insertion is to have both good gates and faulty gates within FPGA as shown left [FPGA95].**

Hardware Approache to Fault Simulation (FPGA)

