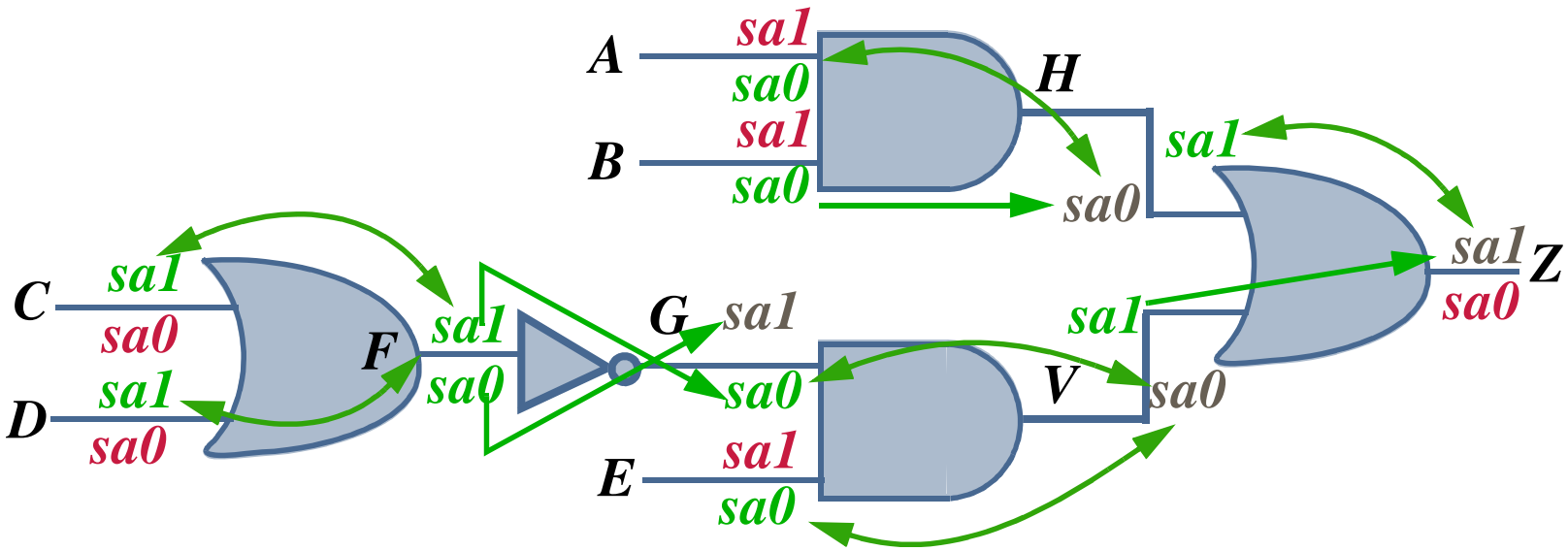
Fault Equivalence Review

Equivalence fault collapsing is performed in a level-by-level pass from inputs to output using *local* (gate level) fault equivalences.



One of the faults in each of this equivalence remains →

{A/0, B/0, H/0}
{C/1, D/1, F/1, G/0}
{E/0, G/0, V/0}
{H/1, V/1, Z/1}
{F/0, G/1}

10/20 = 0.5 remain

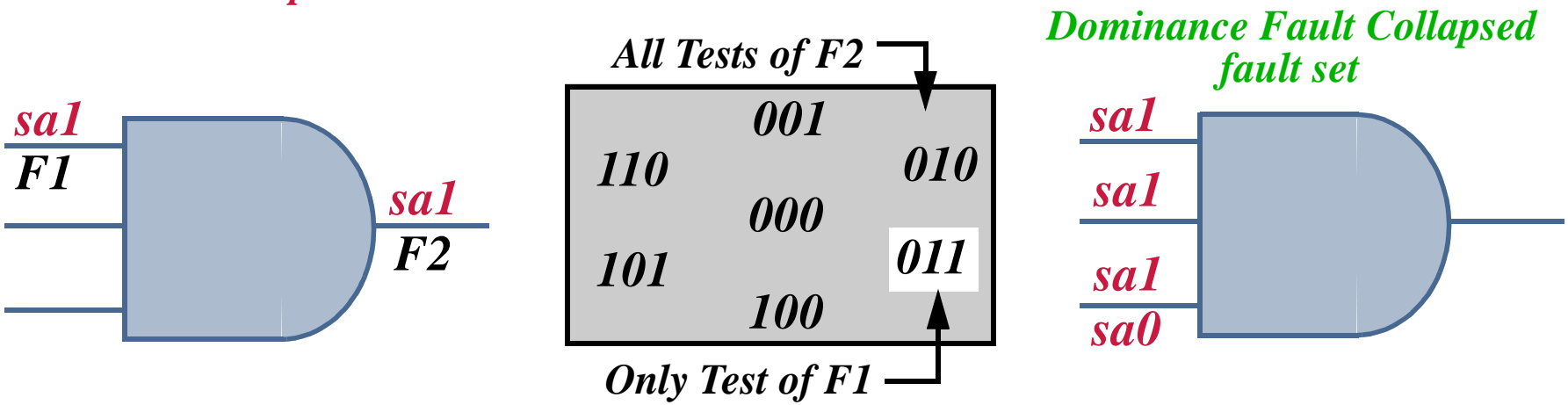
Faults with no equivalence also remain

Reduction is between 50-60% and is larger, in general, for fanout free circuits.

Fault Dominance

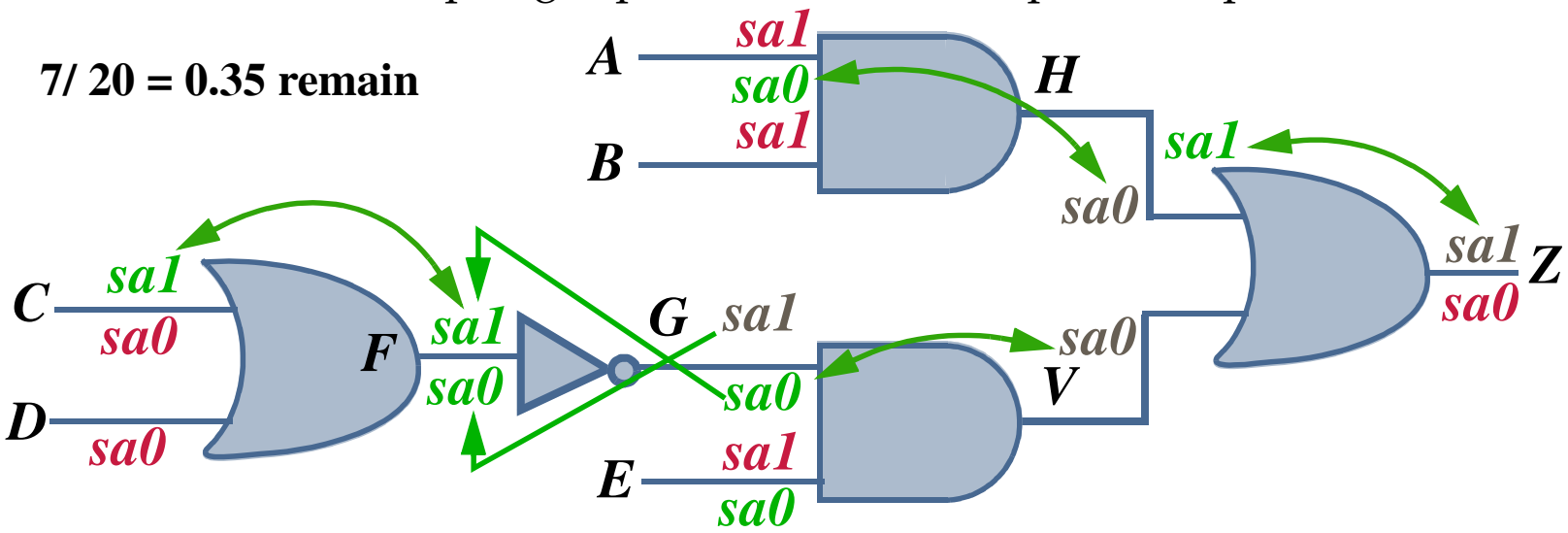
- If fault detection is the objective (not diagnosis), then ***fault dominance*** can be used to further reduce the fault list.
- If all tests of some fault F1 detect another fault F2, then F2 is said to dominate F1.
- Dominance Fault Collapsing: If fault F2 dominate fault F1, then F2 is removed from the fault list.
- When dominance fault collapsing is used, it is sufficient to consider only the input faults of Boolean gates. Illustrated in the next example.
- In a tree circuit (without fanouts) PI faults form a dominance collapsed fault set.
- If two faults dominate each other then they are equivalent.
- For sequential circuits, it should be noted that equivalence fault-collapsing techniques are valid but dominance fault-collapsing techniques are NOT.

Dominance Example



Dominance fault collapsing is performed from outputs to inputs.

7/ 20 = 0.35 remain



One such fault list may be: {*A/0*, *A/1*, *B/1*, *C/0*, *C/1*, *D/0*, *E/1*}

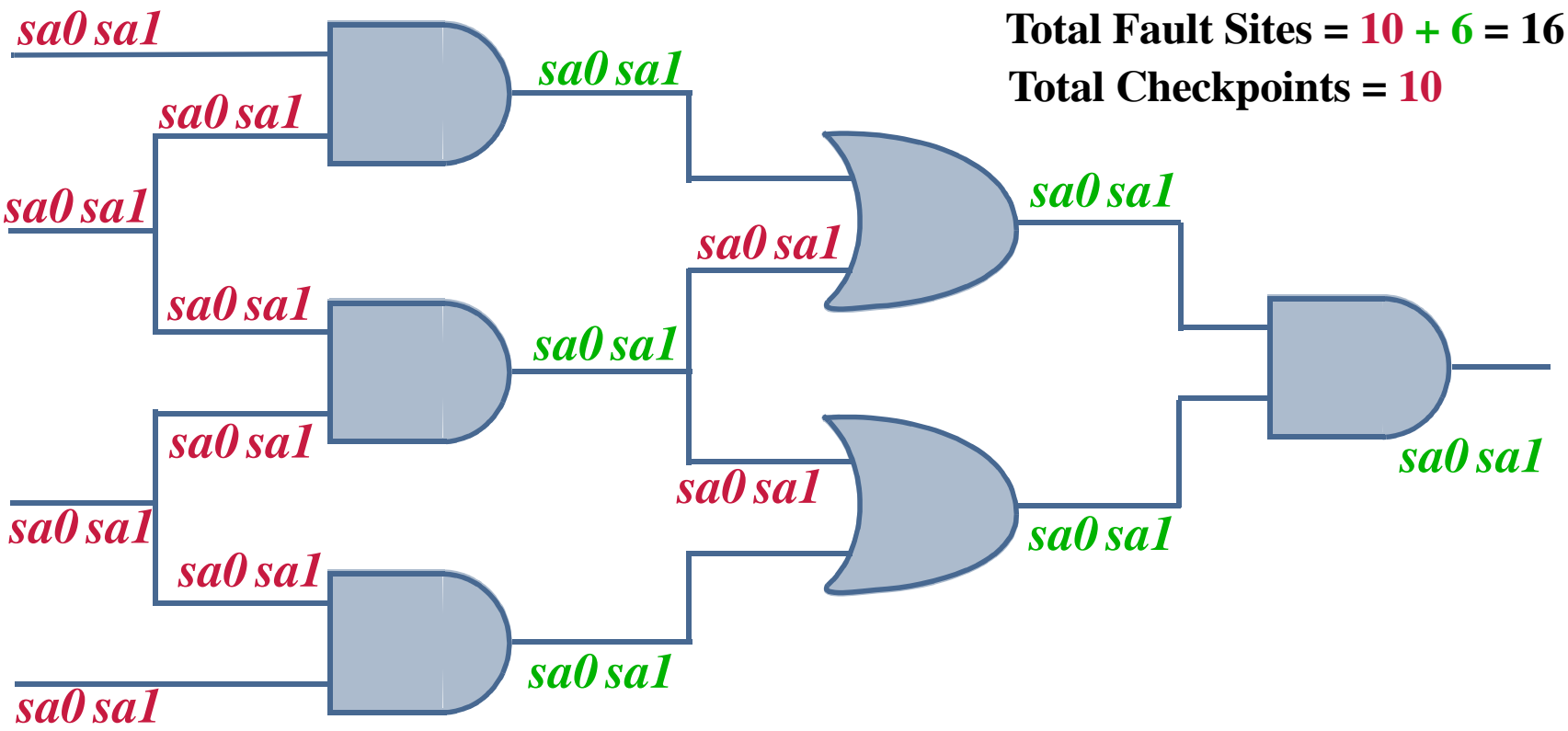
Or another may be: {*B/0*, *A/1*, *B/1*, *C/0*, *D/1*, *D/0*, *E/1*}

Checkpoint faults

Primary inputs and fanout branches of a combinational circuit are called ***checkpoints***.

Checkpoint Theorem

A test set that detects all single (multiple) stuck-at faults on all checkpoints of a combinational circuit, also detects all single (multiple) stuck-at faults in that circuit.



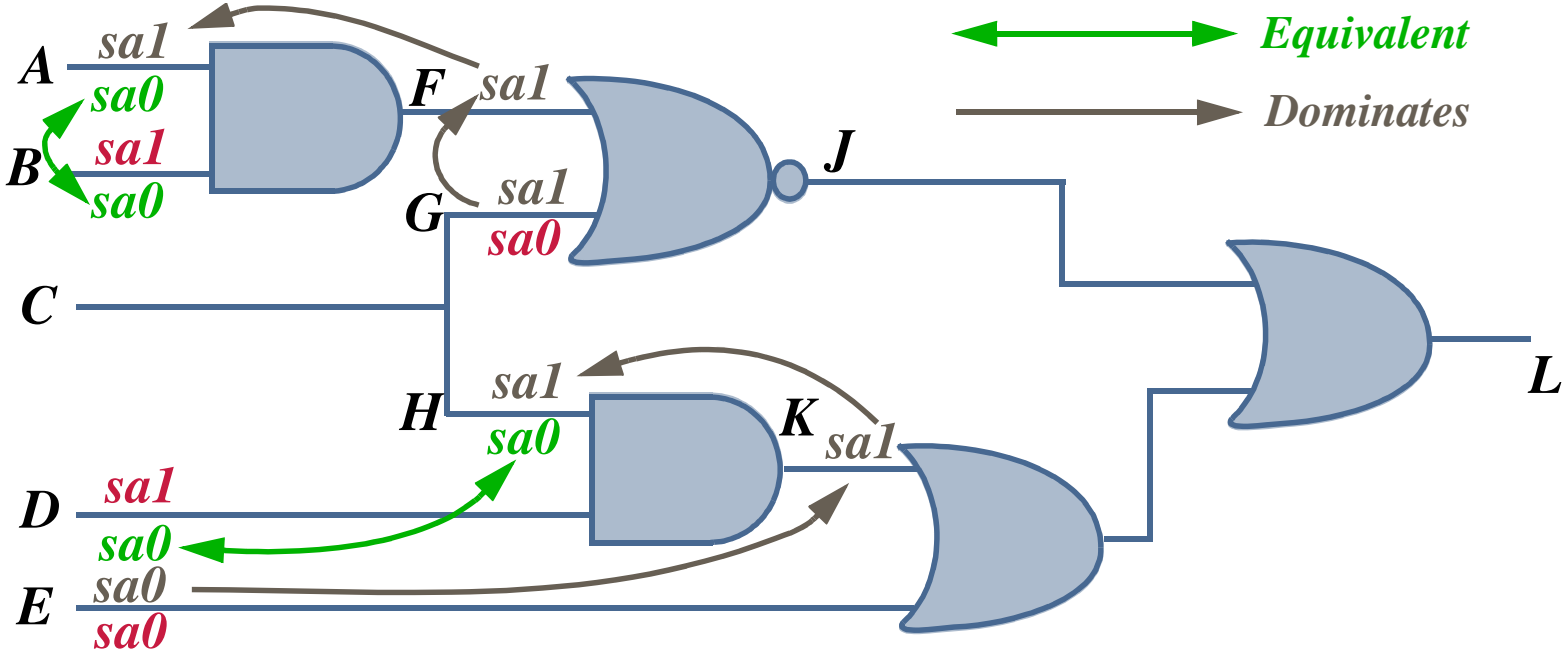
Checkpoint Faults

Therefore, it is sufficient to target faults only at the checkpoints.

Structural equivalence and dominance relations can then be used to further collapse the list of faults.

For example, this circuit has 24 SSFs.

But it only has 14 checkpoint faults (the 5 PIs) + *G* and *H*.



This leaves 10 faults from the original list of 14 checkpoint faults.

Classes of Stuck-At Faults

Following classes of single stuck-at faults are identified by fault simulators:

- Potentially-detectable fault: Test produces an unknown (X) state at primary output (PO); detection is probabilistic, usually with 50% probability
- Initialization fault: Fault prevents initialization of fault circuit; present in circuits with memory elements; can be detected as a potentially detectable fault
- Hyperactive fault: Fault induces much internal signal activity without reaching PO
- Redundant fault: A fault that done not modify the input-output function of the circuit. A redundant fault cannot be detected using SSF tests
- Untestable fault: Faults for which the test generator is unable to find a test.

Multiple Stuck-At faults

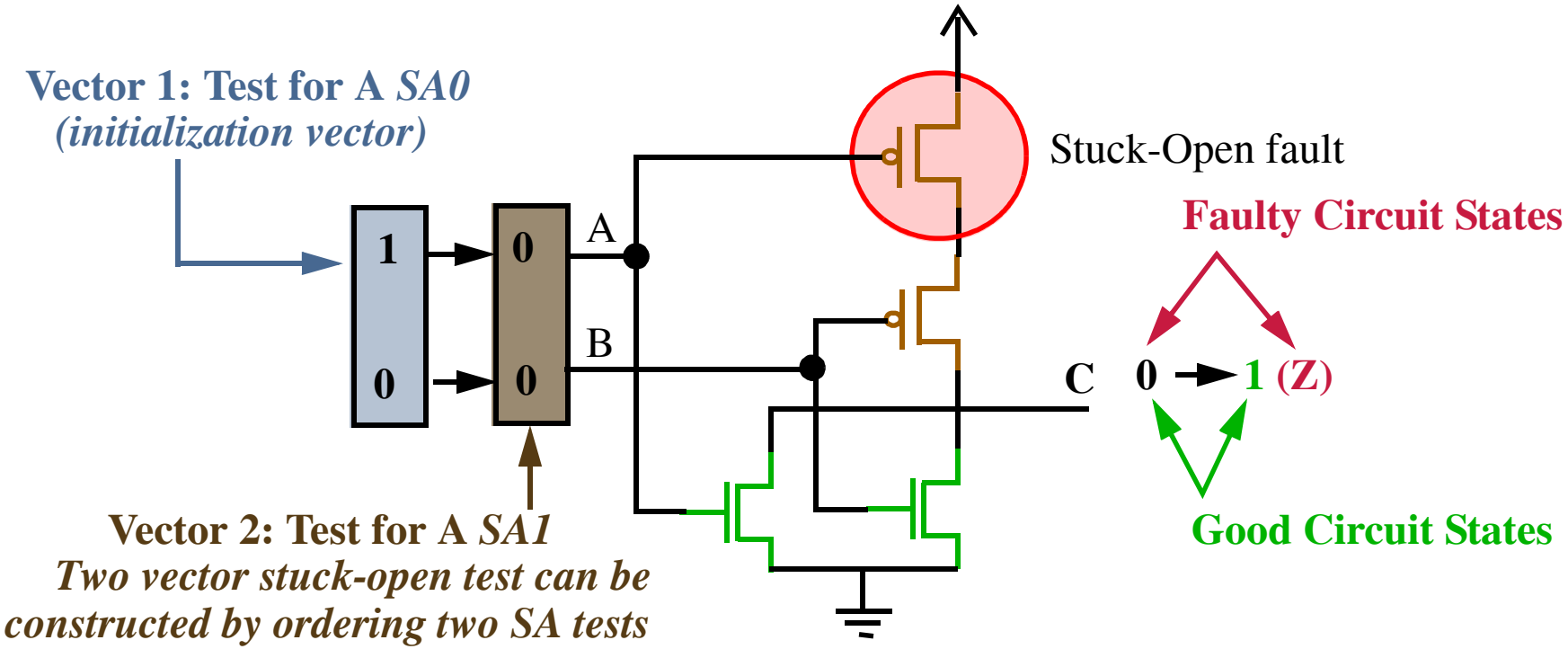
- A multiple stuck-at fault means that any set of lines is stuck-at some combination of (0,1) values
- The total number of single and multiple stuck-at faults in a circuit with k single fault sites is 3^k-1
- A single fault test can fail to detect the target fault if another fault is also present, however, such masking of one fault by another is rare.
- Statistically, single fault tests cover a very large number of multiple faults.
- It might be important to consider them if diagnostic or fault localization procedures don't work with multiple faults.

Transistor (Switch) Faults

MOS transistor is considered as an ideal switch and two types of faults are modeled:

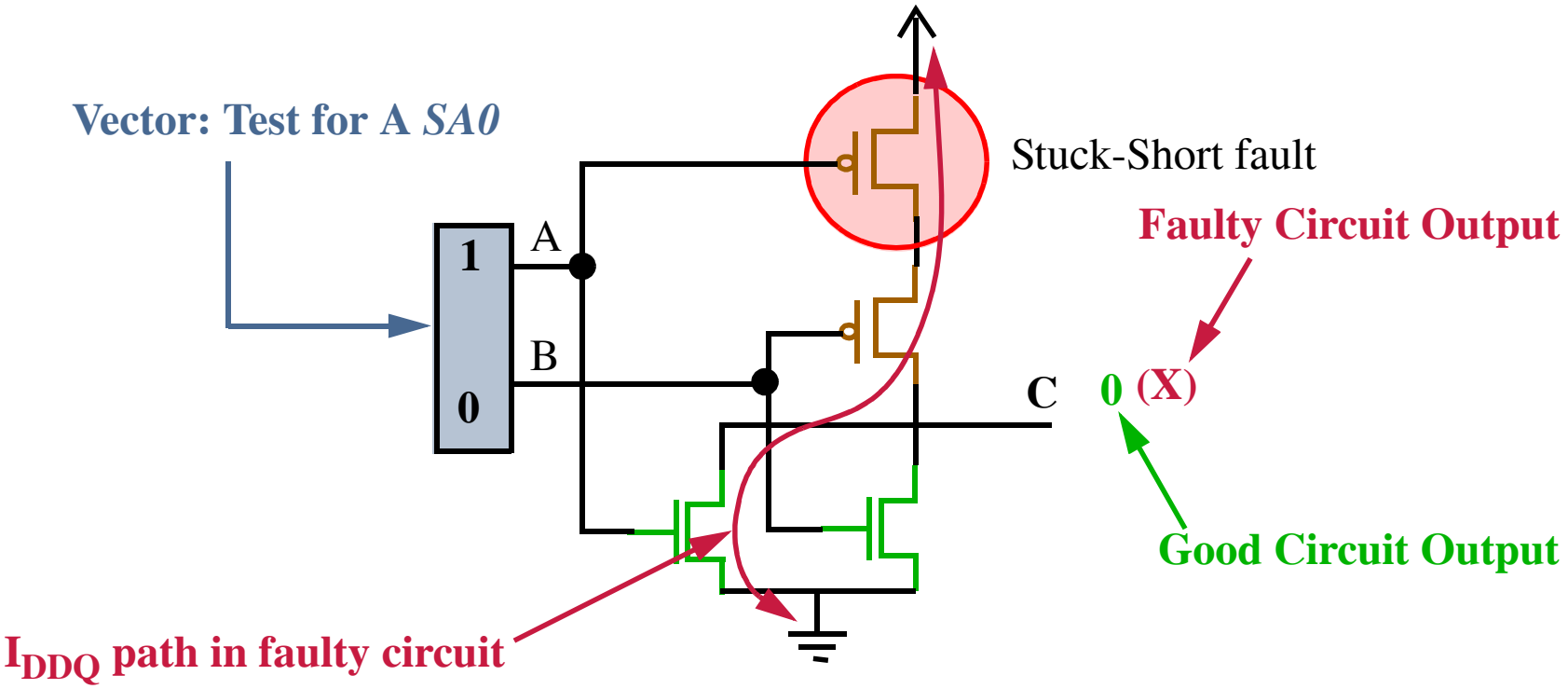
- Stuck-open: a single transistor is permanently stuck in the open state
- Stuck-short: a single transistor is permanently shorted irrespective of it's gate voltage.

Detection of a stuck-open fault requires a two vector sequence.



Transistor (Switch) Faults

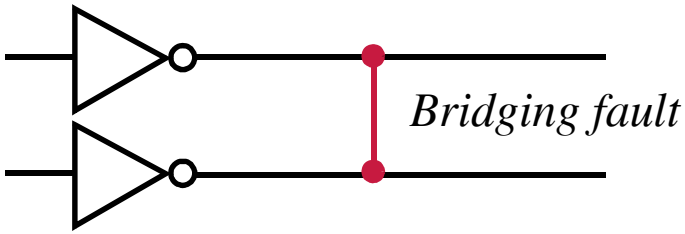
Detection of a stuck-short fault requires the measurement of quiescent current (I_{DDQ}).

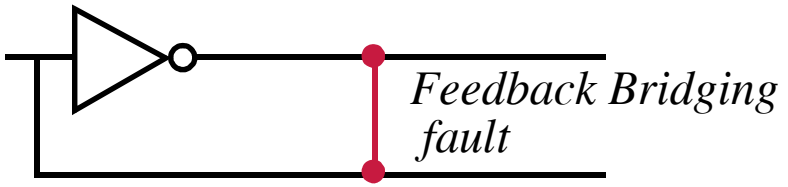


Bridging Faults

Modeled at the gate or transistor level as a short between 2 (simple) or more of ***signal*** lines.

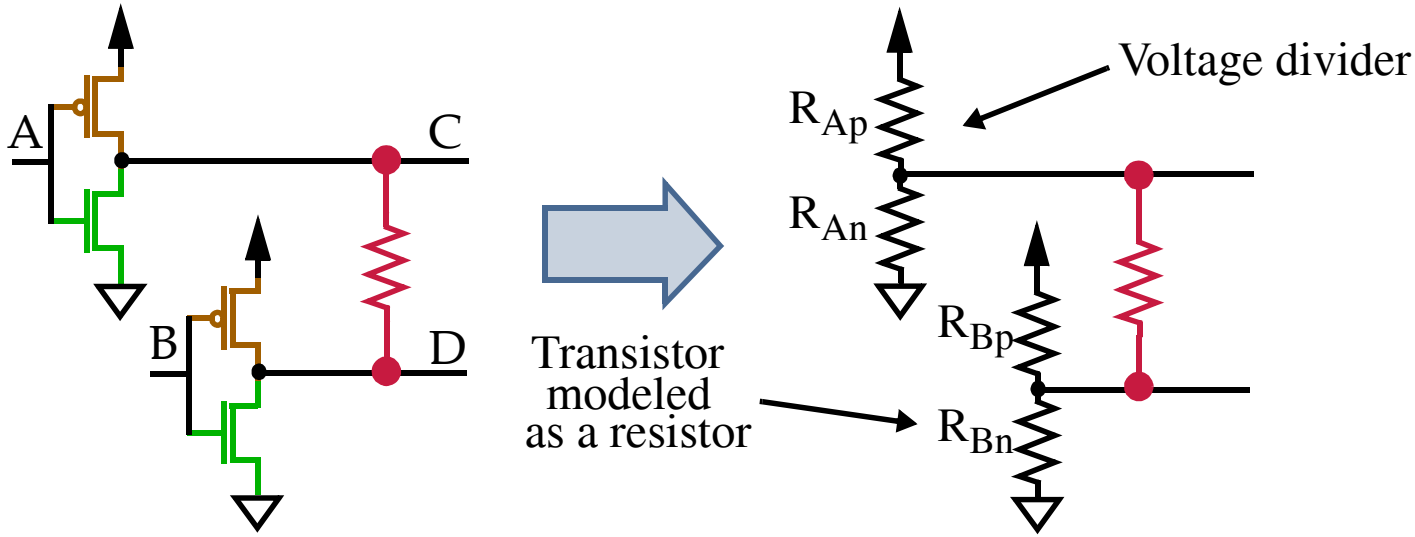
Non-feedback versus feedback (memory) versions.





Fault is usually modeled using ***wired logic***: AND and OR.

For CMOS, it ***depends*** on the *type of gates* driving the shorted lines and their *input values*.

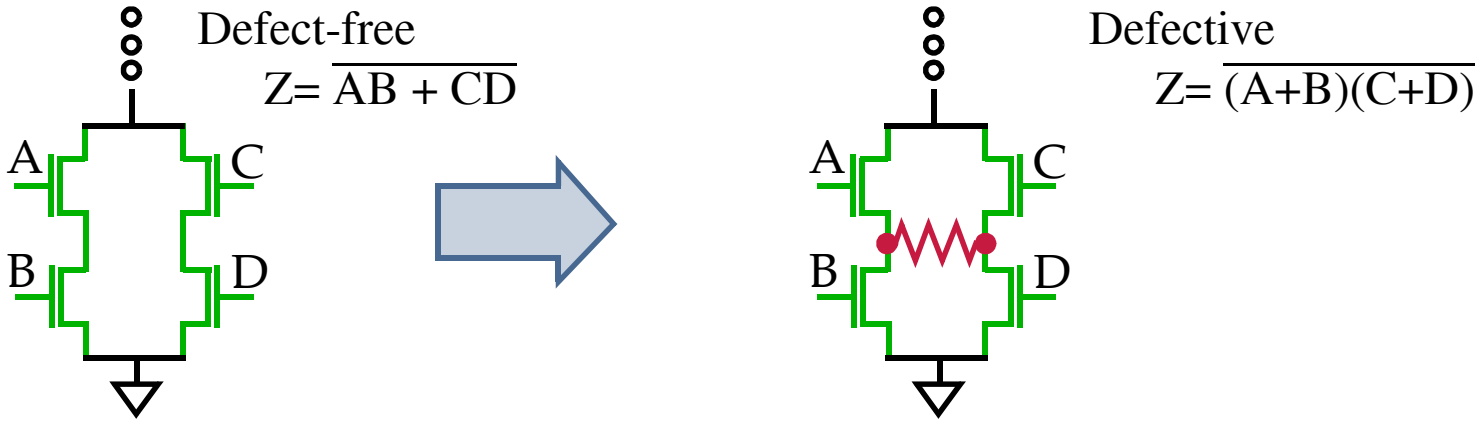


Bridging Faults

The transistor ***resistances*** determine the appropriate model:

Input values	*Resistance relationships*	*Resulting output value*	*Wired logic model.*
A=B	Any ratio	C = D	AND, OR
A=0, B=1	$R_{Ap} > R_{Bn}$	C = D = 0	AND
	$R_{Ap} < R_{Bn}$	C = D = 1	OR
A=0, B=1	$R_{An} > R_{Bp}$	C = D = 1	OR
	$R_{An} < R_{Bp}$	C = D = 0	AND

Bridging faults that can not be represented by a ***known*** fault model.



Some convert ***combination*** circuits to ***sequential*** (feedback bridging).

Delay Faults

- ***Transition Fault (gross-delay faults):***

  Gate delay increased to point where transition does not reach output before end of clock period, even along the shortest path.

- ***Gate-delay Fault:***

  Defect increases input to output delay of a single logic gate.

- ***Line-delay Fault:***

  In contrast to transition fault, a test here must propagate the transition through the *longest sensitizable path*.

- ***Path-delay Fault:***

  This fault causes the cumulative propagation delay of a path to increase beyond some specified time duration.

- ***Segment-delay Faults:***

  A segment (of length L gates) delay fault increases the delay of a segment such that all paths containing the segment have a path-delay fault.

  Segment-delay = Path-delay if L is the maximum combinational depth of the circuit.

  Segment-delay = Transition fault if L is 1.