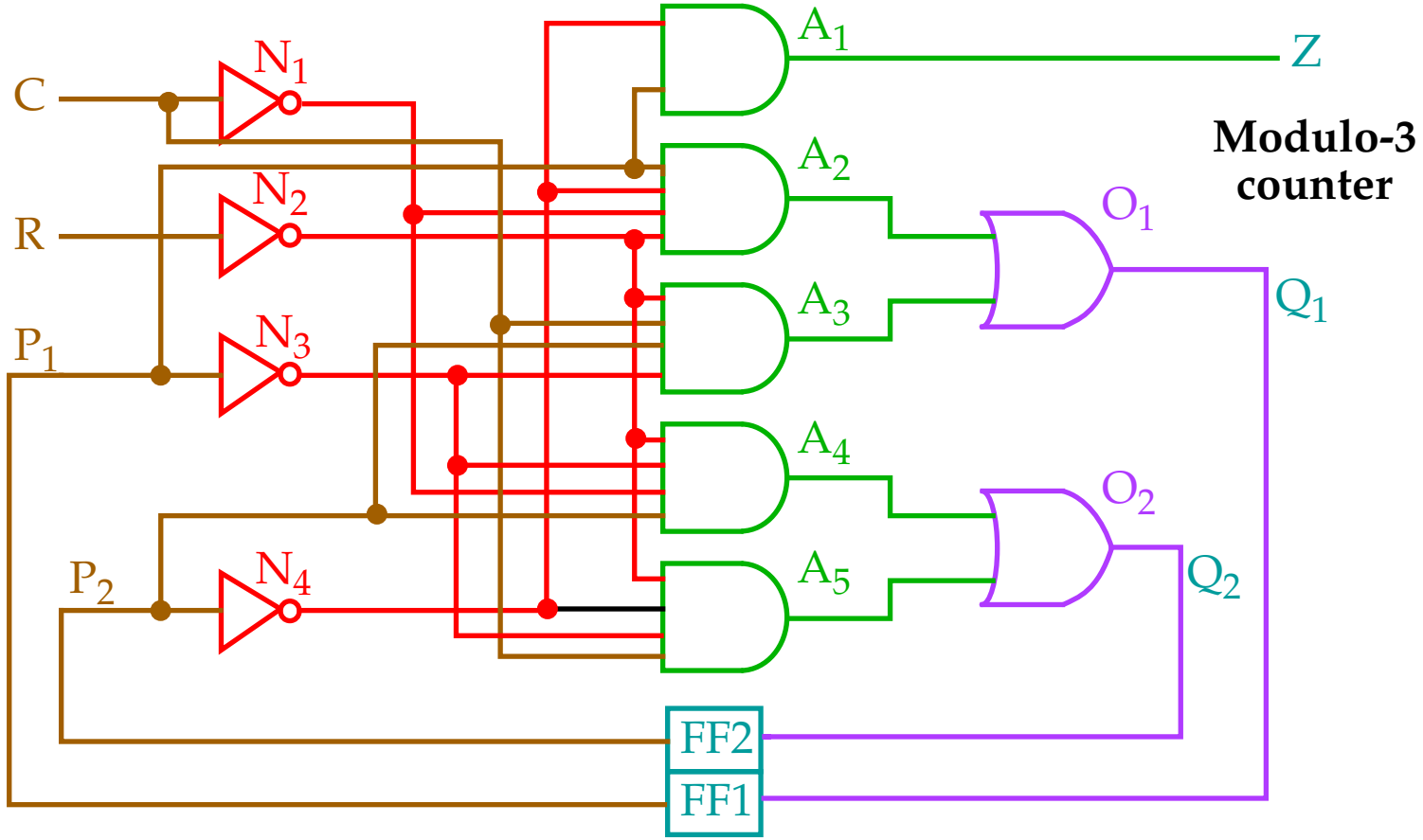
**Path Counting**

The number of paths can be an exponential function of the # of gates.

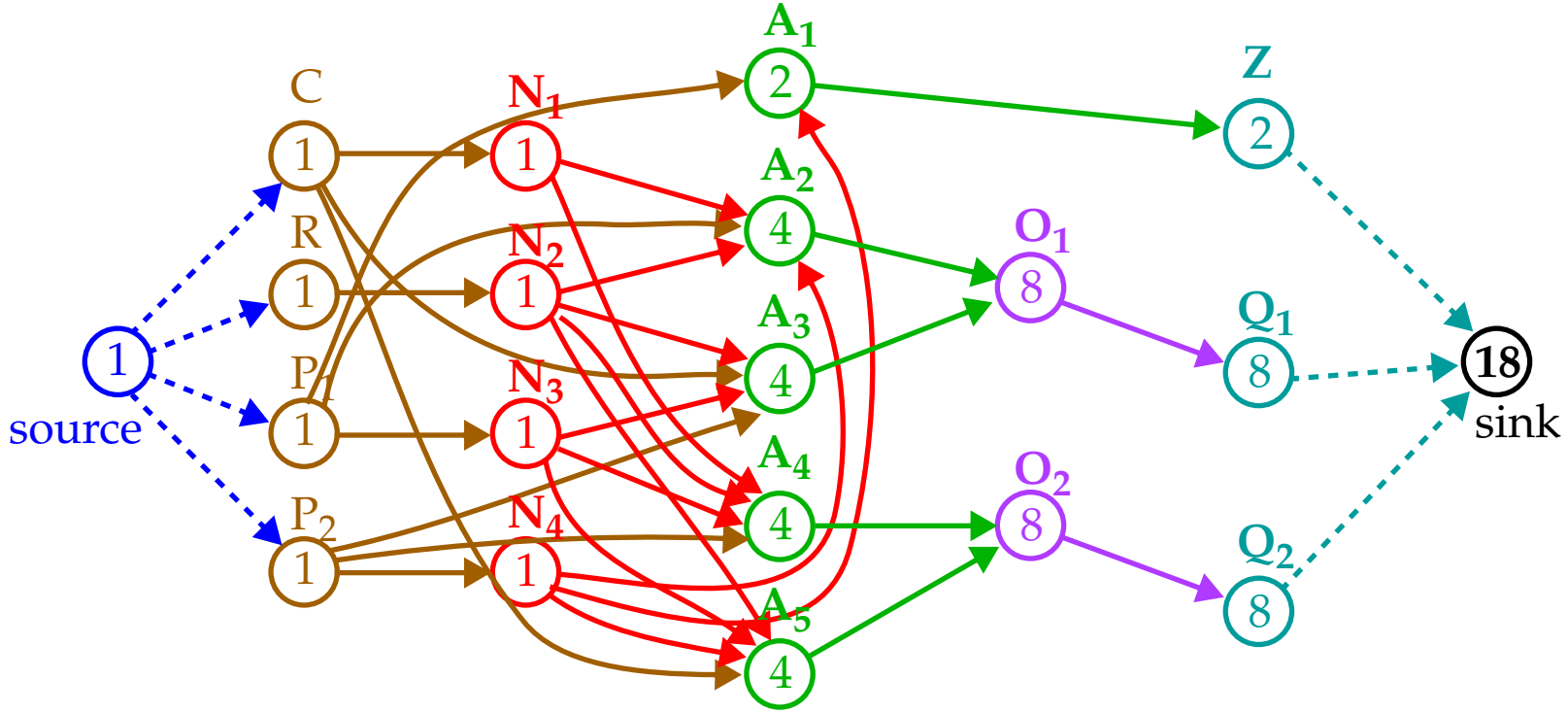
Parallel multipliers are notorious for having huge numbers of paths.

It is possible to efficiently count paths in spite of this however.



**Path Counting**

*Directed acyclic graph* (DAG) path graph for this circuit:




Vertices represent the PIs, POs and gates, edges represent signal flow.

Source and sink nodes are added to point to PIs and POs, algorithm visits each node, follows its edges and adds src value to destination nodes.

Since the maximum *indegree* is O(N), worst case complexity is $O(N^2)$.

**Transition Fault Model**

Faults are modeled at the gate I/Os as *slow-to-rise* (STR) and *slow-to-fall* (STF) faults that elicit **Stuck-At** type fault behavior at the POs.

For detection of a *slow-to-rise* fault, start with a *SA0* fault on the line.

- This sets the line to 1 and propagates the state of the line to a PO.
- Let this be vector $V_2$ then define $V_1$ as a vector that sets the line to 0.

Advantages include:

- Number of tests is upper bounded by twice the number of lines.
- Stuck-at ATPG algorithms can be easily modified to produce these tests.

Transition fault tests can detect large (gross) delays.

Tested paths may be short.

They are not reliable at detecting delay defects that are **distributed**, unlike PDF.

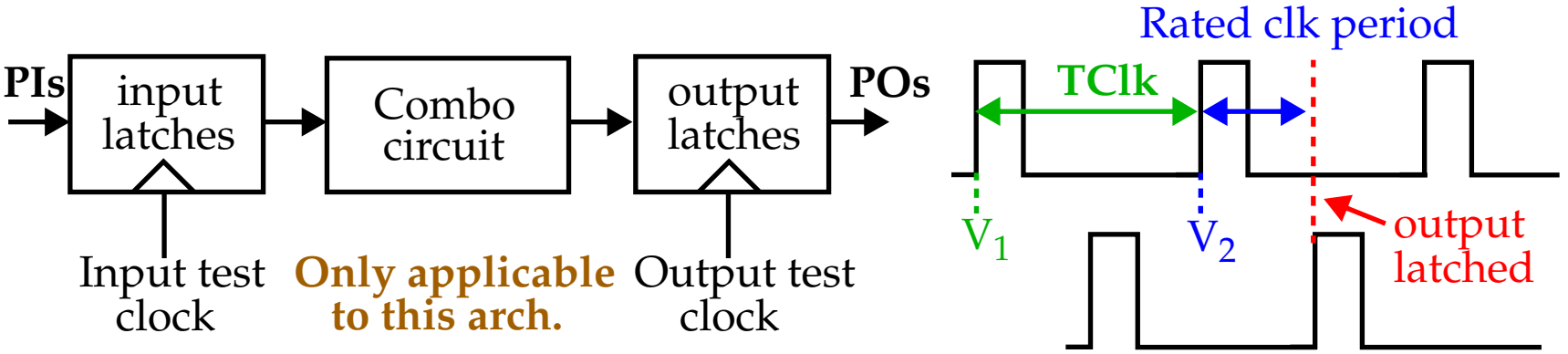
Transition fault tests are usually augmented by *critical* path delay tests.

**Delay Test Methodologies**

The application of delay tests depends on the type of circuit and the DFT hardware used.

- Slow-clock combinational test
- Enhanced-scan test
- Normal-scan sequential test
- Variable-clock non-scan sequential test
- Rated-clock non-scan sequential test

**Slow-clock combinational test**



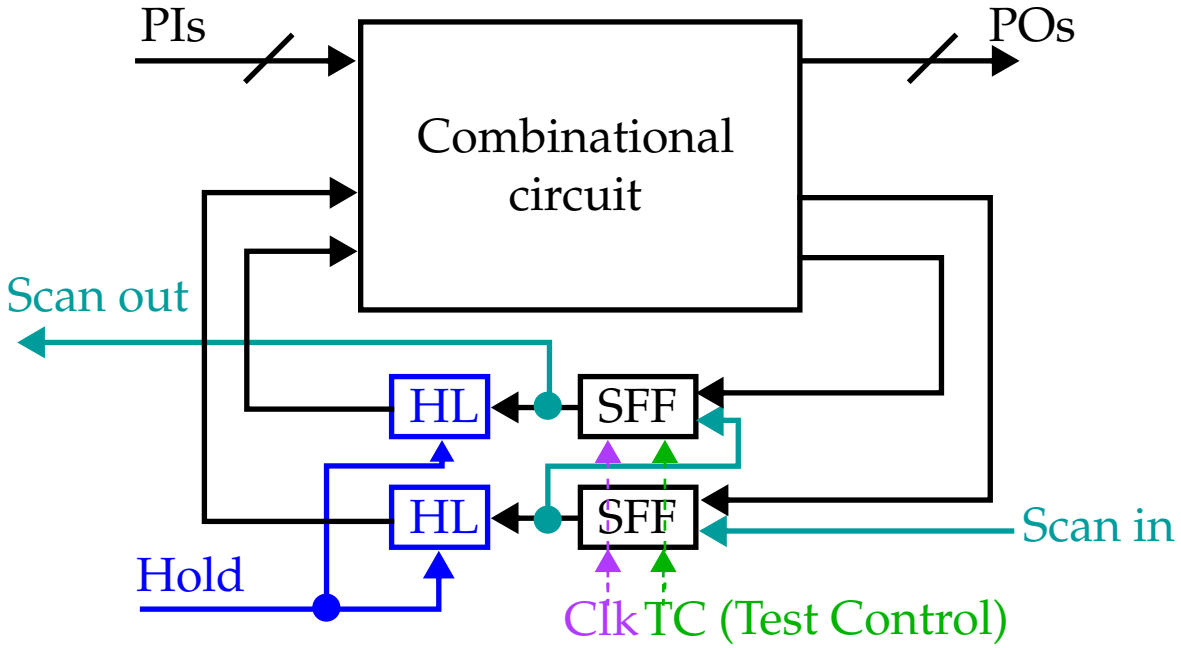

Note that $V_1$ is applied at a slower rate and the circuit is allowed to stabilize.

**Delay Test Methodologies**

**Enhanced scan test**

Applicable to scan types of sequential circuits.

Similar to the previous method, any arbitrary vector pair can be applied and test generation can treat the circuit as combinational.




Each vector consists of two parts, bits for the PIs and bits for the state variables (SFFs).

State bits are scanned in by setting TC to 0 and applying Clk.

**Delay Test Methodologies**

**Enhanced scan test** (cont.)

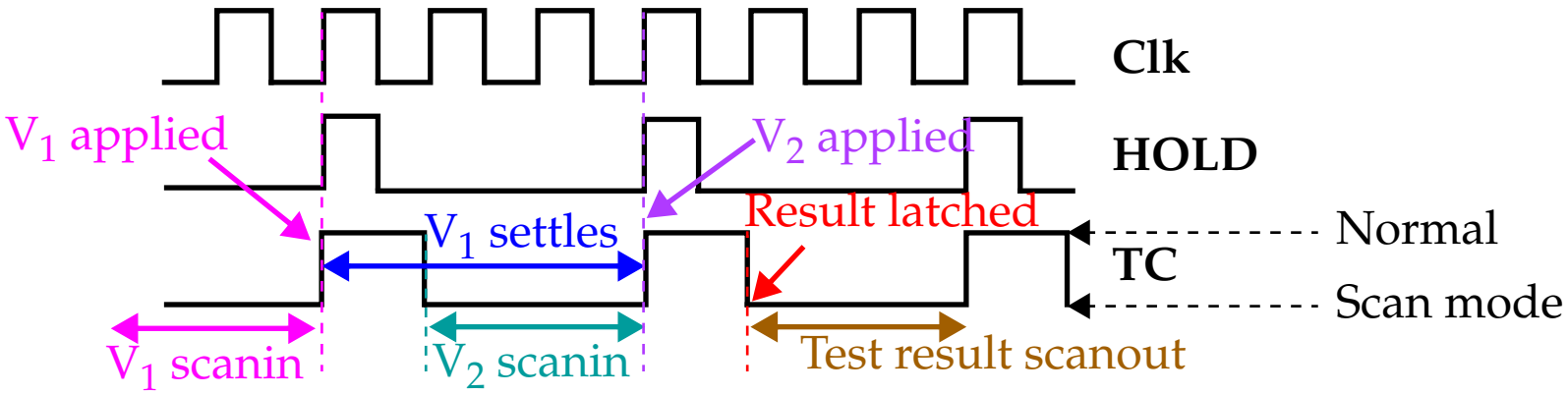
The bits are often scanned in using a slow clock to reduce power consumption and the chance of errors occurring due to scan chain delays.

The scanned $V_1$ bits are transferred to the *Hold Latches* (HL) and the PI bits of $V_1$ are applied.

When $V_1$ stabilizes, the state bits of $V_2$ are scanned in.

Activation of the *Hold* signal and application of the $V_2$ bits to the PIs creates the $V_1$ -> $V_2$ transition.

With TC = 1, Clk is used to latch the outputs in *normal mode*.



## Delay Test Methodologies

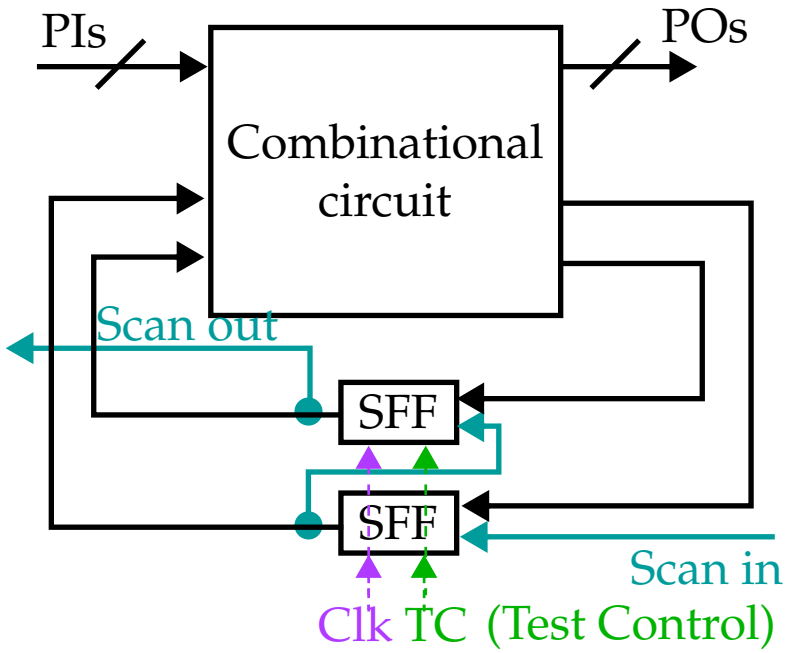
**Enhanced scan test** (cont.)

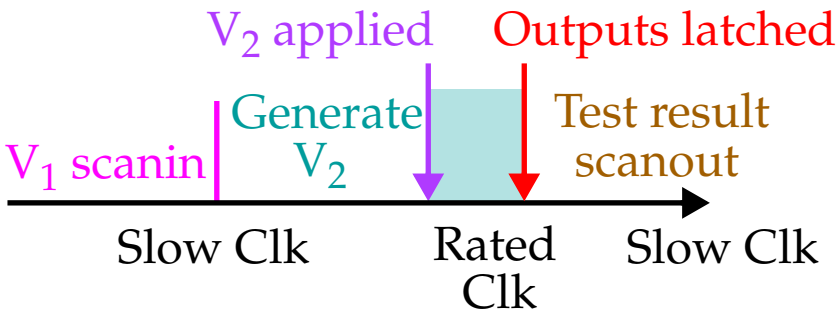
Scan test time similar to full scan design but scan area overhead is larger and *Hold Latches* increase delay in signal paths.

**Normal-scan sequential test**

It is still possible to test full scan circuits with no *Hold Latches* for delay faults.

However, it requires special vector-pairs.




*Scan-shift delay test* or
*Skewed-load delay test*

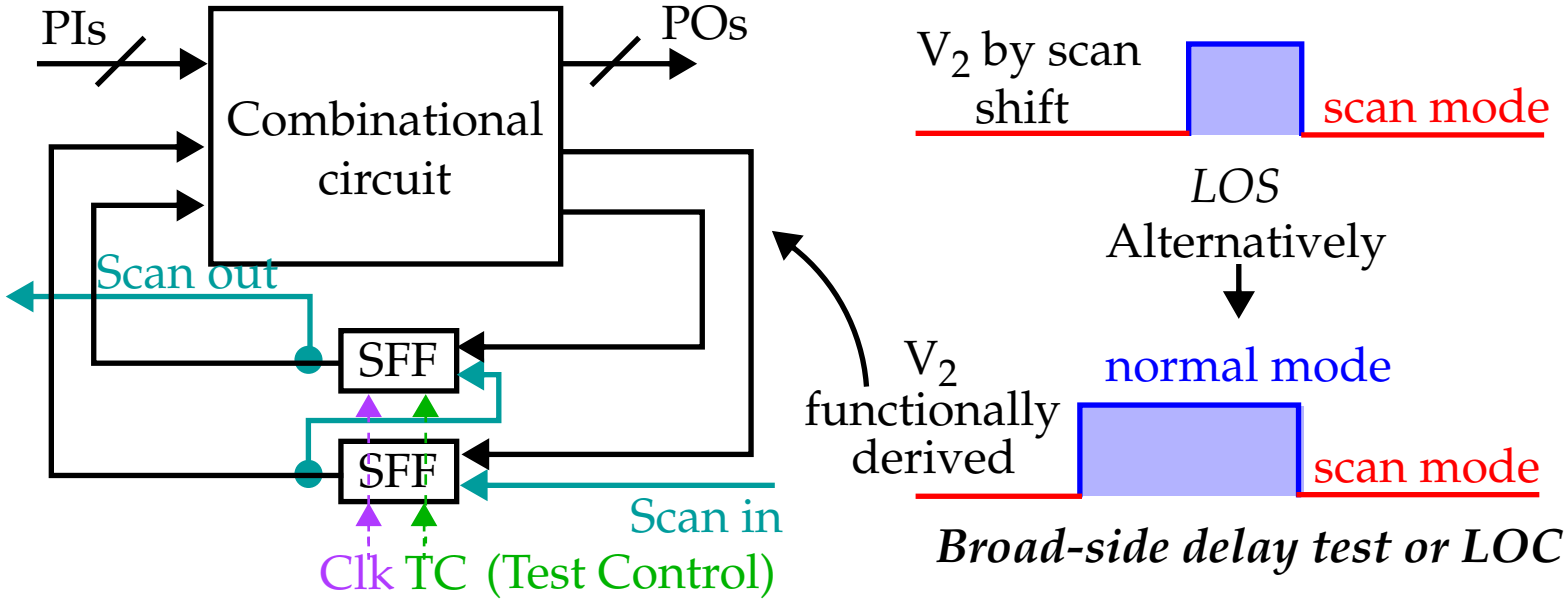
**Delay Test Methodologies**

**Normal-scan sequential test**

*Launch-on-shift (LOS) or scan-shift delay test*: Scan in of $V_1$ is followed by one extra cycle of slow clock with the circuit still in scan mode (TC = 0).

The test is designed so that $V_2$ is obtained from $V_1$ by a 1 bit translation (PI bits of both vectors are unrestricted).

As soon as $V_2$ is applied, mode is changed from scan to normal and Clk is controlled at the rated period to latch outputs.



**Delay Test Methodologies**

**Normal-scan sequential test**

*Launch-on-capture (LOC)* or *broad-side* delay test, the **state** portion (FF values) of $V_2$ are functionally generated by the combo logic under $V_1$.

Simultaneous application of $V_2$ at the PIs and into the FFs via Clk in normal mode generates the $V_1$ -> $V_2$ transitions.

The outputs are latched one rated clock period later.

Disadvantages:

For LOS, scan-enable must switch at rated speed of clk.

For LOC, correlations between $V_1$ and $V_2$ may not allow high fault coverage.

See text for *Variable-clock non-scan sequential test* and *rated-clock non-scan sequential test*

**Practical Considerations in Delay Testing**

Today, verification requires both function and timing analysis.

*Static timing analysis* examines combinational paths without regard to sensitization (delays of gates and wires are looked up in a database).

Results of timing analysis used to improve the design and test:

- *Timing simulation*:
  Identified critical paths are simulated and the design is "tweeked" to make sure it meets the timing specification.

- *Critical path tests*:
  Critical path delay determines the clock period, and therefore tests are usually included to test such paths.

- *Layout optimization*: Critical path data is used for std. cell/custom block placement, to establish priorities in routing and for transistor sizing.

**Practical Considerations in Delay Testing**

**Critical path** tests are good at detecting "correlated defects", i.e., slow-downs due to global process variations, because the longest paths will fail first.

**Spot defects** (or gross defects) affect only a small number of paths in the chip.

Transition fault tests are capable of detecting these gross delay defects.

Two forms of **at-speed** testing:

- External:

  The combination of *critical path* testing and *transition fault* testing provides adequate at-speed testing.

- Built-in self-test:

  Since the at-speed ATE is expensive, **BIST** is an alternative.

  On-chip hardware is needed for test generation and response analysis.

  The speed of BIST is controlled by the off-chip clock.