



Application-Dependent Testing of FPGAs

Mehdi Baradaran Tahoori

PhD Candidate, Stanford CRC

MSEE'02 Stanford, BSCE'00 Sharif

Research Interests: FPGA Testing,

Reliability in DSM VLSI

Background

- Problem Statement
 - ◆ Test routing resources used in a mapped Design
- Motivation
 - ◆ Mapped designs not fully testable
- Applications
 - ◆ Application-Specific FPGA
 - ◆ System-level Testing

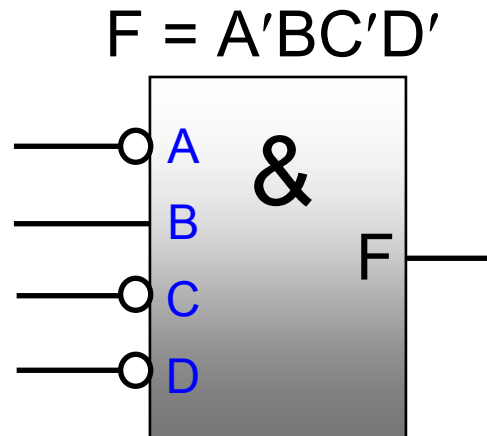
Approach

- Implement **testable** functions in logic blocks
 - ◆ Configuration of routing resources unchanged
- **Advantages**
 - ◆ No extra place-and-route for test configurations
 - ◆ Fast reconfiguration time
 - ◆ Configuration storage overhead minimized
 - ◆ No fault missed due to partitioning
 - ◆ Works for all SRAM-based FPGAs and CPLDs

Basic Idea

- **Single-term** functions
 - ◆ Logic function with only one minterm or maxterm
- **Activating Input**
 - ◆ input values for single minterm (maxterm)
- ★ All sensitized faults are detectable
 - ◆ Test vector = activating input
 - ◆ $A1, B0, C1, D1, F0, A_{BF}B, B_{BF}C, B_{BF}D$

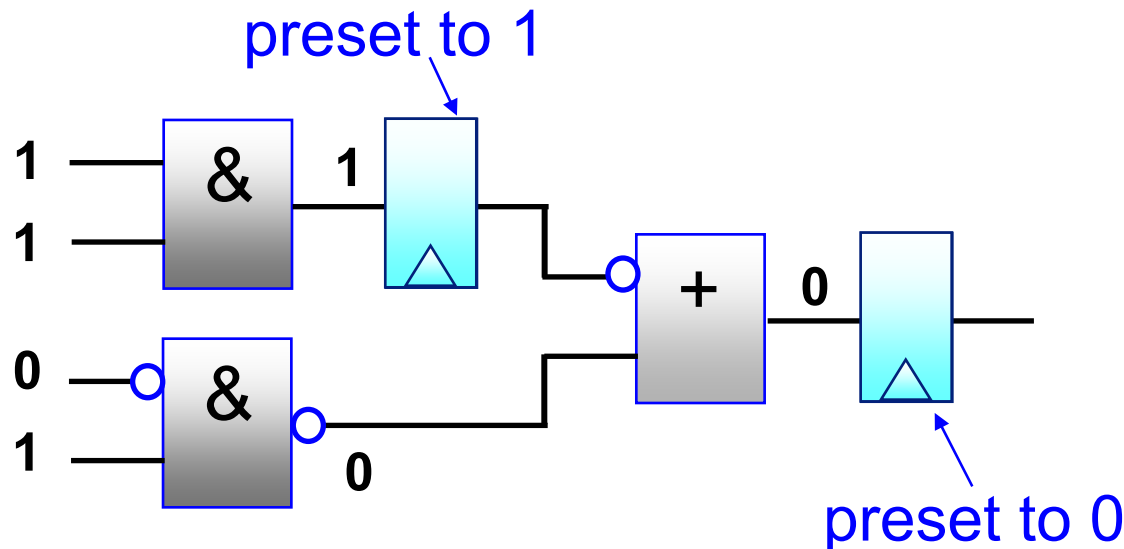
B stuck-at 0



Bridging between B and D

Entire FPGA

- Single-term functions in all used logic cells
 - ◆ Inputs of each logic cell = activating input
- Sequential Designs
 - ◆ Preset flip-flops
 - ◆ Number of test cycles = Max ***sequential depth***
 - ☞ Longest sequential path from PI to PO



Test Configuration Generation

- Which single-term functions?

- ◆ Depend on fault list

- ☞ Transition faults : all-AND, all-OR

- ◆ Robust transition fault testing

- ☞ Bridging faults

- ☞ Other...

