V. Digital System Testing

5. Digital System Testing

5.1 Introduction to Digital System Testing

5.2 Fault and Fault Models

5.3 Boundary Scan

5.1 Introduction to Digital System Testing

- Highlight of IC
  - LSI (1000), VLSI (100000) ULSI (1M)
  - Circuit densities: doubling every 18 month
  - Reliability up – better testing
  - Cost Down
  - Yield up
- Testing
  - Main purpose of fault testing is the detection of malfunctions
  - Location of the fault may also be desired

5.1 Introduction to Digital System Testing (cont'd)

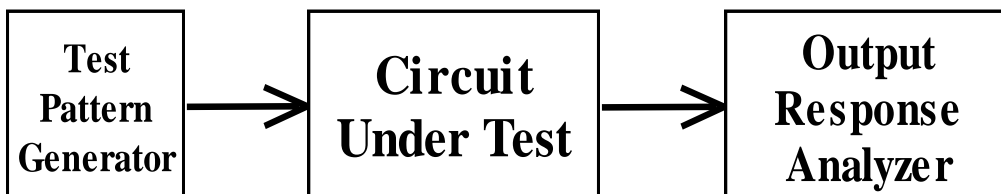
- Why test IC
  - To verify design
  - To detect faults arising from manufacture or ware-out
  - To ensure components meet design specification (parameter)
- Cost of Testing
  - IC $X
  - PCB board $ 10X
  - System $ 100X
  - Test cost > 50% production cost

5.1 Introduction to Digital System Testing (cont'd)

- Verification
  - A design is checked to meet the requirement and specifications
  - Functions test checks that the circuit implement the functions that is required, and does not implement the function that is not required
- Factors
  - Original design specifications
  - Delays, hazards
  - Routing, layout design rules
- Tools
  - Simulators : logic, timing, functions behavioral
  - HDL, VHDL

5.1 Introduction to Digital System Testing (cont'd)

- Testability
  - Observability: the internal signals can be determined at the output
  - Controllability: producing an internal signal value by applying signals to inputs
- Testing method
  - Off-line: external + BIT(built in test)
  - On-line: BIT (built-in test)
  - Example
- Structural testing

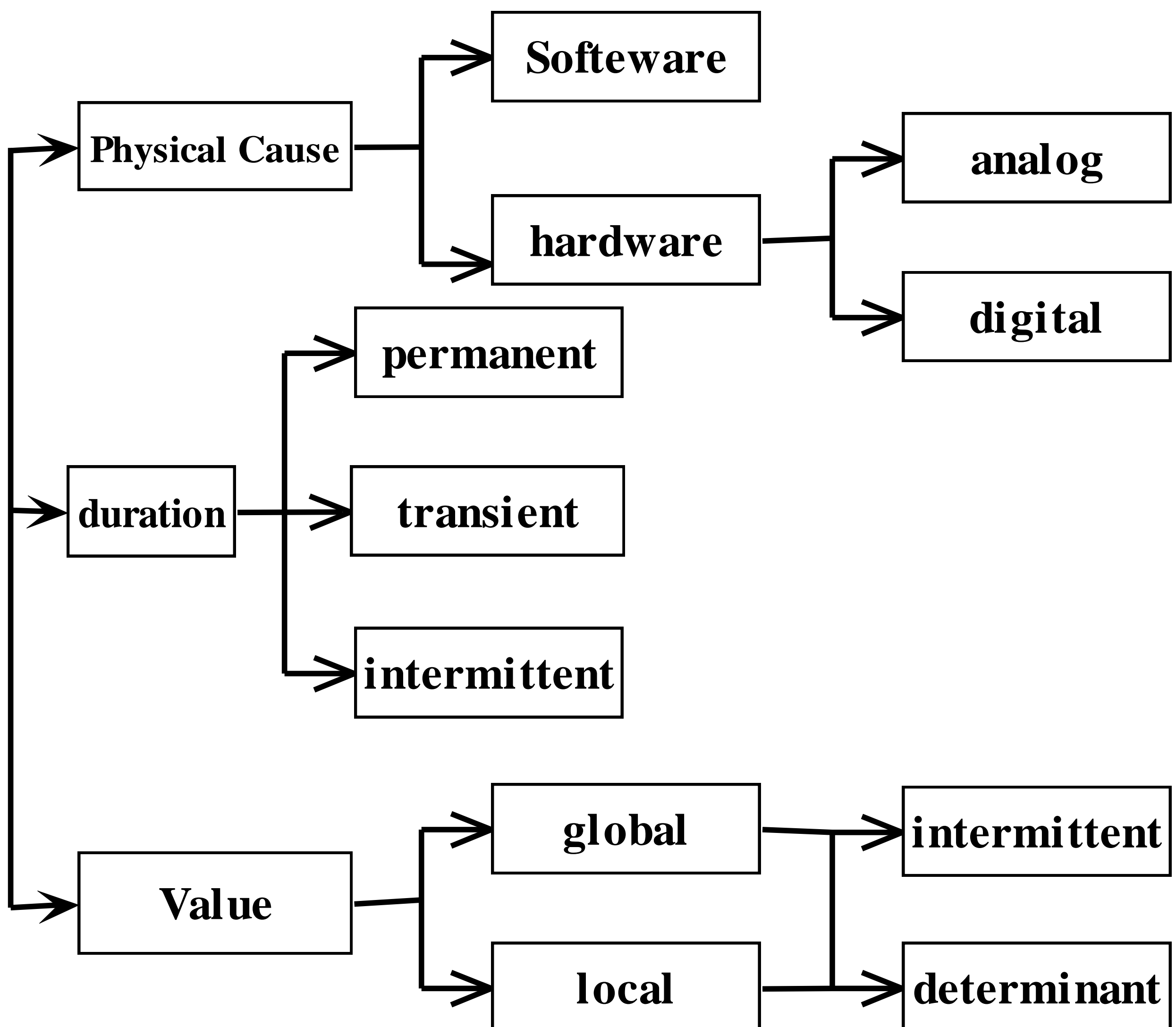


5.1 Introduction to Digital System Testing (cont'd)

- Ideal Testing Environment
  - Store the truth table of the circuit
  - Apply all possible input combinations
  - Examine the output responses
- Real testing environment
  - Fault modeling
    - Model physical detects by a small number of faults
  - Test pattern generation
    - Generate a test pattern
  - Output response analysis
    - Analyze the circuit output without storing al the circuit responses
- Fault coverage
  - Number of detected faults/ number of faults modeled

5.2 Fault and Fault Models

- Before testing takes place, it is necessary to examine failure mechanisms, their effects and methods of modeling them

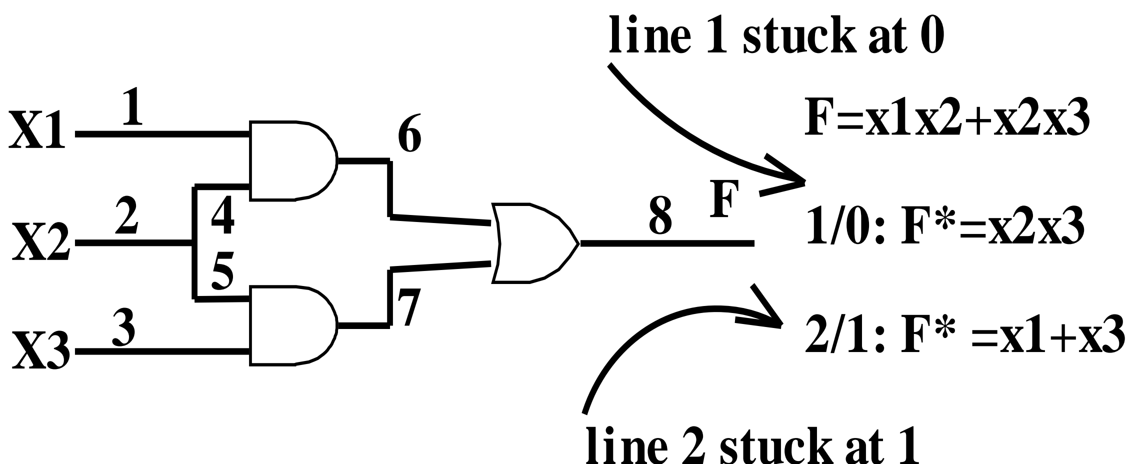


5.2 Fault and Fault Models (cont'd)

- Fault characteristics:
  - Permanent fault– fault in existence long enough to be observed at test time
  - Transient fault: fault appears and disappears in short intervals of time
  - Intermittent fault: fault which appears and disappears at regular time intervals
- Faults models
  - A logical abstraction describing the functional effects of a physical failure
  - Different levels of modeling are used based on different primitives
  - Block: functional model, block diagram (data-path)
  - Gate: switching level, gates and latches
  - Circuit: electrical model, use transistors, resistors capacitors
  - Geometrical-model: layout description of chip ..

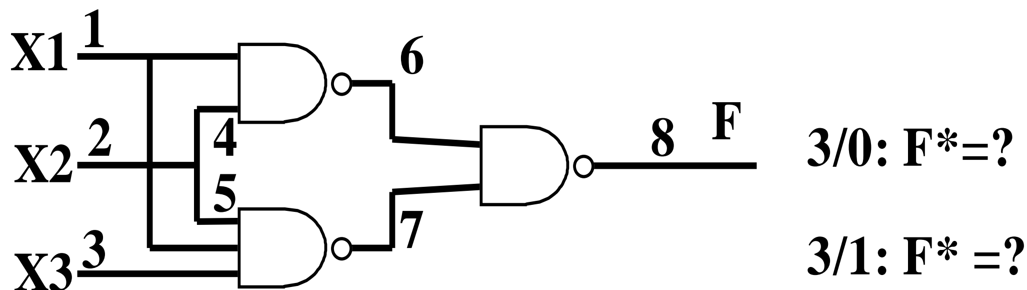
5.2.1 Stuck at Fault Model

- Most commonly used fault model
- May consider single or multiple stuck-at-faults
- The effect of the fault is modeled by having a line segment stuck at 0 or 1
  - All gates are perfect, problem may occur on line segments.
  - Single stuck at fault (SSF)
  - Example:
  - {x1,x2,x3} : inputs {1 2 3,4,5,6,7,8} : line segment number, F : correct output function, F* incorrect output function



5.2.1 Undetectable Faults

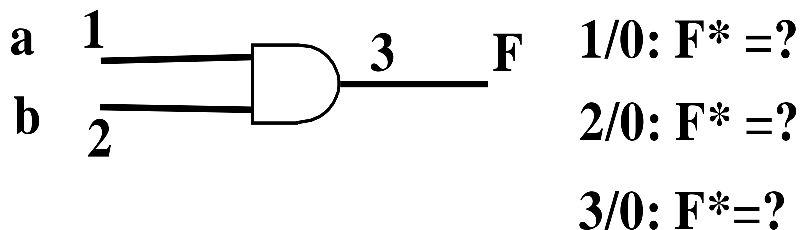
- Undetectable fault: its effect cannot be seen at the output, no matter what test pattern is applied



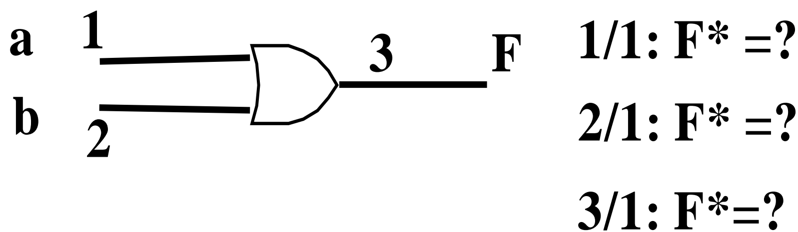
- Problem: redundancy may lead to undetectability
  - Why redundancy?
  - How to detect?
  - Undetectable faults may lead to problems in detecting other detectable faults

5.2.1 Equivalent Faults

- What is equivalent faults?
- Example1:

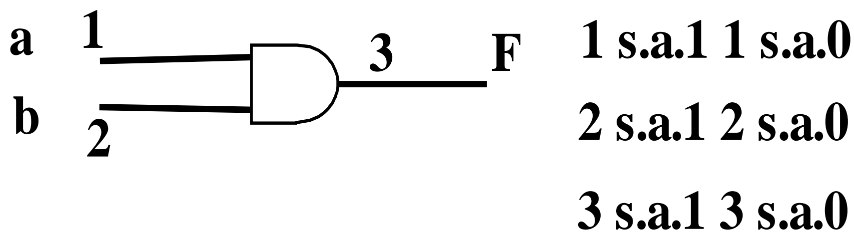


- Example 2:

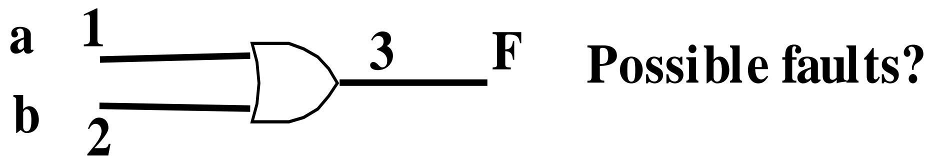


5.2.1 Equivalent Faults (cont'd)

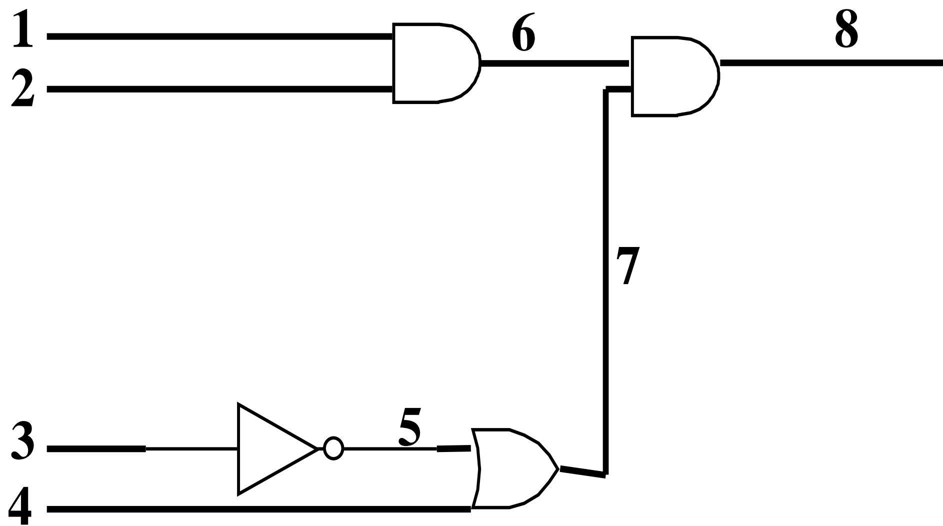
- What is equivalent faults?
- Example1:



- Example 2:

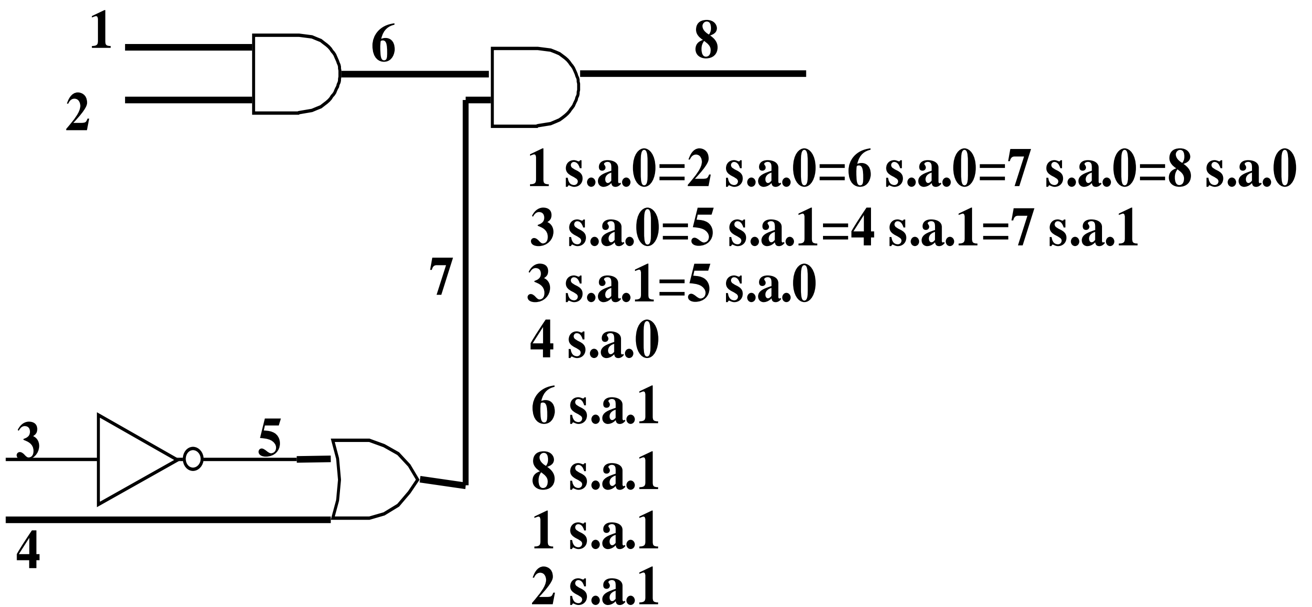


5.2.1 Equivalent Faults (cont'd)



5.2.2 Determine Test Pattern Generation

- Find a test vector for a given fault
  - Fault propagation
  - Path sensitization and backtracking

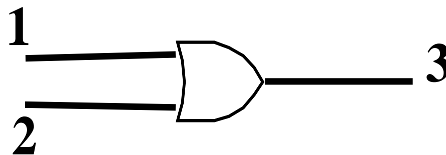


  - 1/0 : 1234= 1101,1111,1100
  - 3/0 : 1234= 1110
  - 3/1 : 1234= 1100
  - 4/0 : 1234 = 1111
  - 6/1 : 1234 =
  - For each of the test faults, find the test vectors
  - Remove the redundant one, find the minimum set

5.2.2 Determine Test Pattern Generation (cont'd)

Fault equivalence depends on individual gates

- AND any input/0 ←→output/0
- Or any input/1 ←→output/1
- NAND any input/0 ←→output/1
- NOR any input/1 ←→output/0
- NOT any input/0 ←→output/1
- any input/1 ←→output/0



5.2.2 Determine Test Pattern Generation (ATPG)

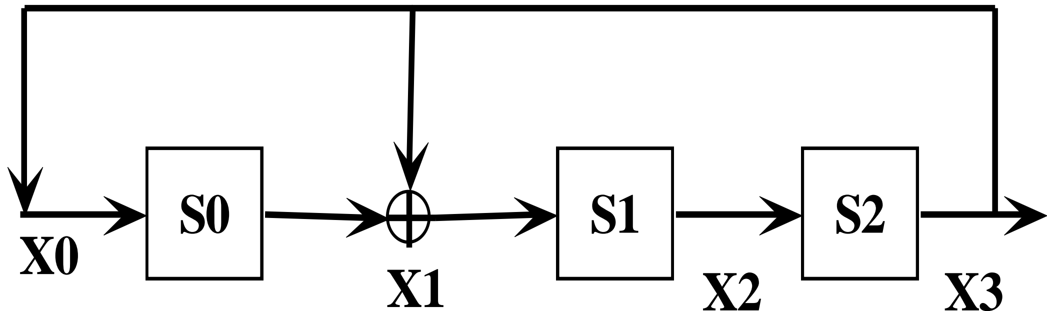
- Given a circuit of n line segments
  - There are 2n single stuck at faults
  - Fault collapsing reduces 2n faults to k faults
  - For each of the k faults, find a test vectors
  - Find a minimal test set from the test vectors that detect the k faults

- Advantages:
  - Short test length
  - 100% fault coverage

- Disadvantages:
  - very computational expensive
  - Need to store test vectors

5.2.3 Pseudorandom Test Pattern Generation

- Using an autonomous linear feedback shift register (ALFSR)



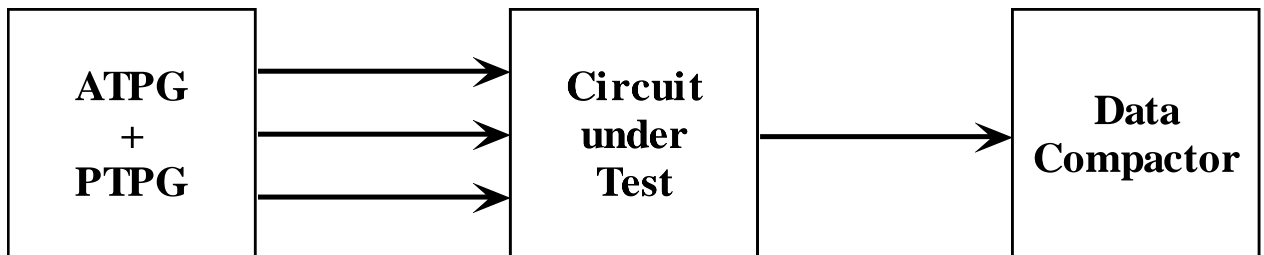
$$P(X)=X^3+X+1$$

Time	So	S1	S2
T1	0	0	1
T2	1	1	0
T3	0	1	1
T4	1	1	1
T5	1	0	1
T6	1	0	0
T7	0	1	0

5.2.3 Pseudorandom Test Pattern Generation (cont'd)

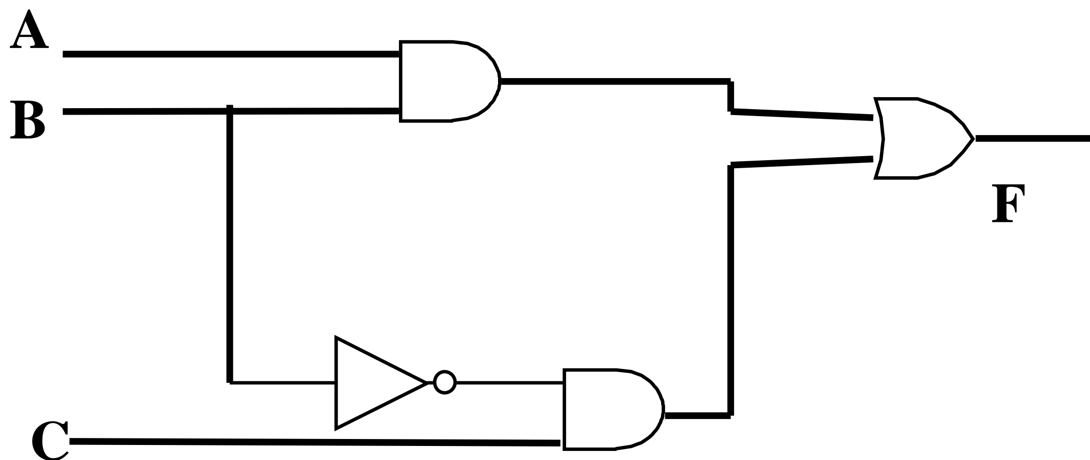
- Given a degree n P(X) in binary field
  - Degree n→ n stage ALFSR
  - If an ALFSR cycles through all 2^n-1 non zero states, it has the maximum length cycle, the corresponding P(x) is primitive. Otherwise, it is non primitive
  - Example :
  - $P(x)=X^7+X^3+X^2+1$
  - Pseudorandom Test Pattern Generation (PTPG)
  - Advantages: easy to generate
  - No need for storage
  - Suitable for built-in self test
  - Disadvantages:
  - Relative long test length
  - Some faults may not be tested?
- Solution
  - DO PTPG cover 90-95%
  - DO ATPG cover rest

5.2.4 An Off-line Test (signature analysis)



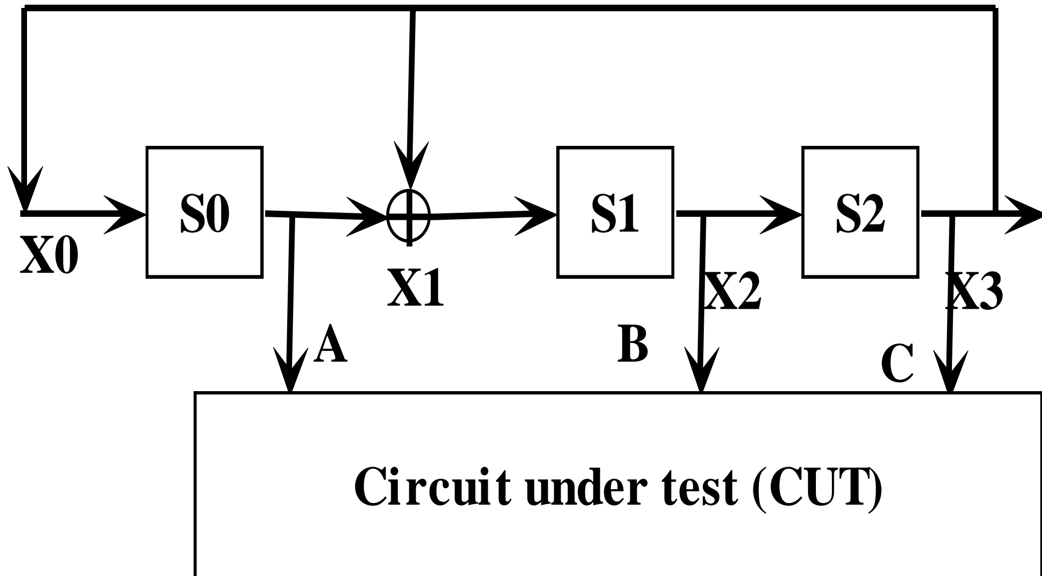
5.2.4 An Off-line Test (Cont'd)

- Example



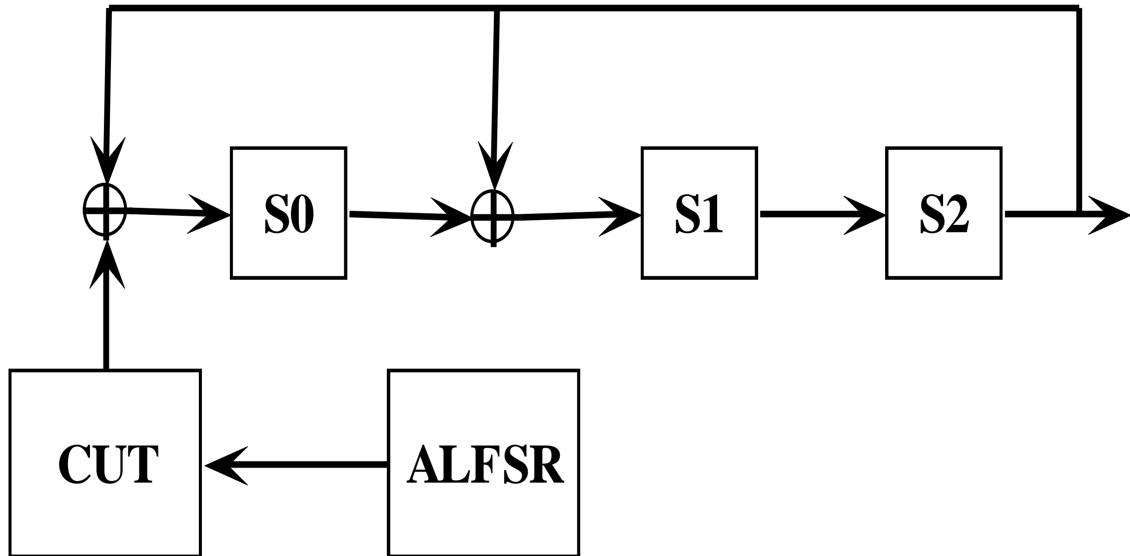
$F = AB + \bar{B}C$

5.2.4 An Off-line Test (Cont'd) example



Time	So	S1	S2	f
T1	0	0	1	1
T2	1	1	0	1
T3	0	1	1	0
T4	1	1	1	1
T5	1	0	1	1
T6	1	0	0	0

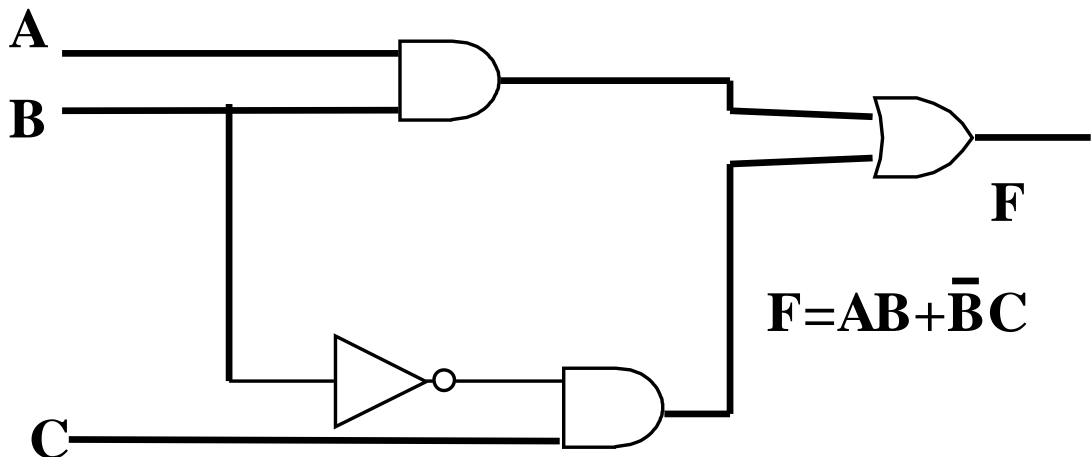
5.2.4 An Off-line Test (Cont'd) example



Time	f	So	S1	S2
T0	--	0	0	0
T1	1	1	0	0
T2	1	1	1	0
T3	0	0	1	1
T4	1	0	1	1
T5	1	0	0	1
T6	0	1	1	1

- At T6, s0s1s2=111 is a fault free signature

5.2.4 An Off-line Test (Cont'd) example



5 s.a.1 **$F^*=A+\bar{B}C$**

Time	So	S1	S2	f
T1	0	0	1	1
T2	1	1	0	1
T3	0	1	1	0
T4	1	1	1	1
T5	1	0	1	1
T6	1	0	0	1

5.2.4 An Off-line Test (Cont'd) example

- At T6, s0s1s2=011 is a fault signature
- Fault is detected.

Time	f	So	S1	S2
T0	--	0	0	0
T1	1	1	0	0
T2	1	1	1	0
T3	0	0	1	1
T4	1	0	1	1
T5	1	0	0	1
T6	1	0	1	1

5.3 Boundary Scan

- A design and test technique that allows circuits to be tested via the board edge connectors
  - I/O hardware is modified in design to incorporate a programmable shift register
  - Overhead minimal as it uses some empty space around I/O pads
- Problems of board test
  - Increasing IC complexity
  - Increasing use of surface-mount technology

5.3 Boundary Scan (cont'd)

- Inclusion of programmable shift register stage adjacent to each I/O pin such that signals at component boundaries can be controlled and observed using scan testing principles