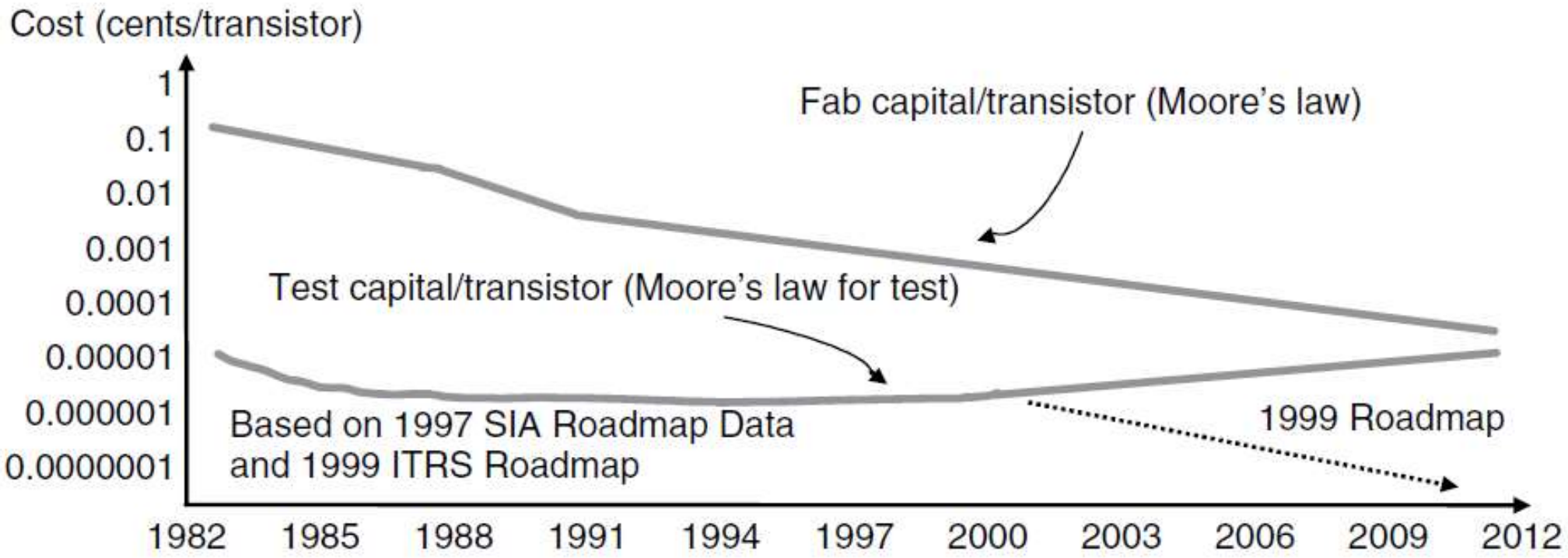
Course Outline

- Semiconductor Industry and Technology Overview
- IC Design Flows
- Timing in Digital Systems
- Front-end Design Flow
- Back-end Design Flow
- Interconnection and Signal Integrity
- Low-Power Design
- **Design-for-Testability (DFT)**

Fabrication vs. Test Cost



Cheng 2006

Yield Loss

$$yield = \frac{number\ of\ acceptable\ parts}{total\ number\ of\ parts\ fabricated}$$

- **Two types of yield loss**
  - Catastrophic yield loss is due to random defects, and
  - Parametric yield loss is due to process variations
    - Parametric is becoming the dominant of the two.

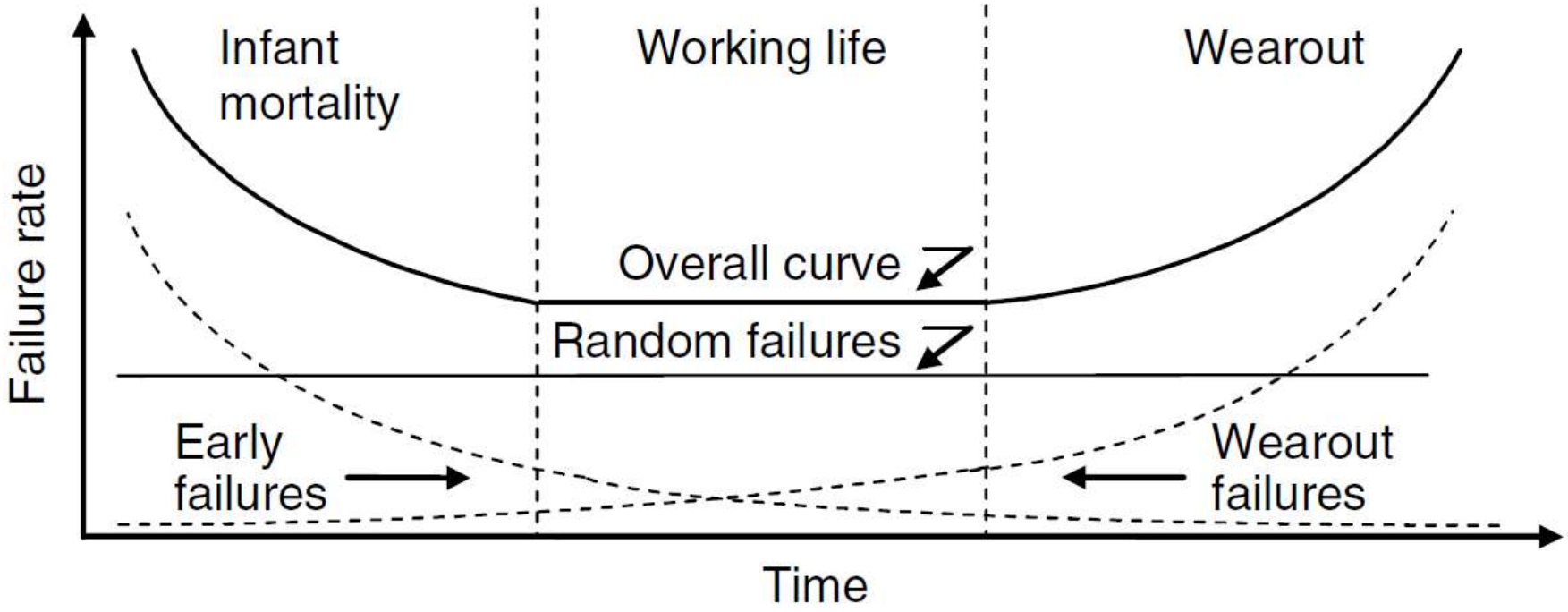
DFY and DFM

- **Methods to reduce process variations during fabrication are generally referred to as design for yield (DFY).**

- **Circuit implementation methods to avoid random defects are generally referred to as design for manufacturability (DFM).**
  - Broadly speaking, any DFM method helps to increase manufacturing yield and thus can be considered as a DFY method.

Failure Modes

- **Manufacturing yield relates to the failure rate.**



$$reject\ rate = \frac{number\ of\ faulty\ parts\ passing\ final\ test}{total\ number\ of\ parts\ passing\ final\ test}$$

Defect levels

$$DL = 1 - Y^{(1-FC)}$$

where Y is yield and FC is fault coverage

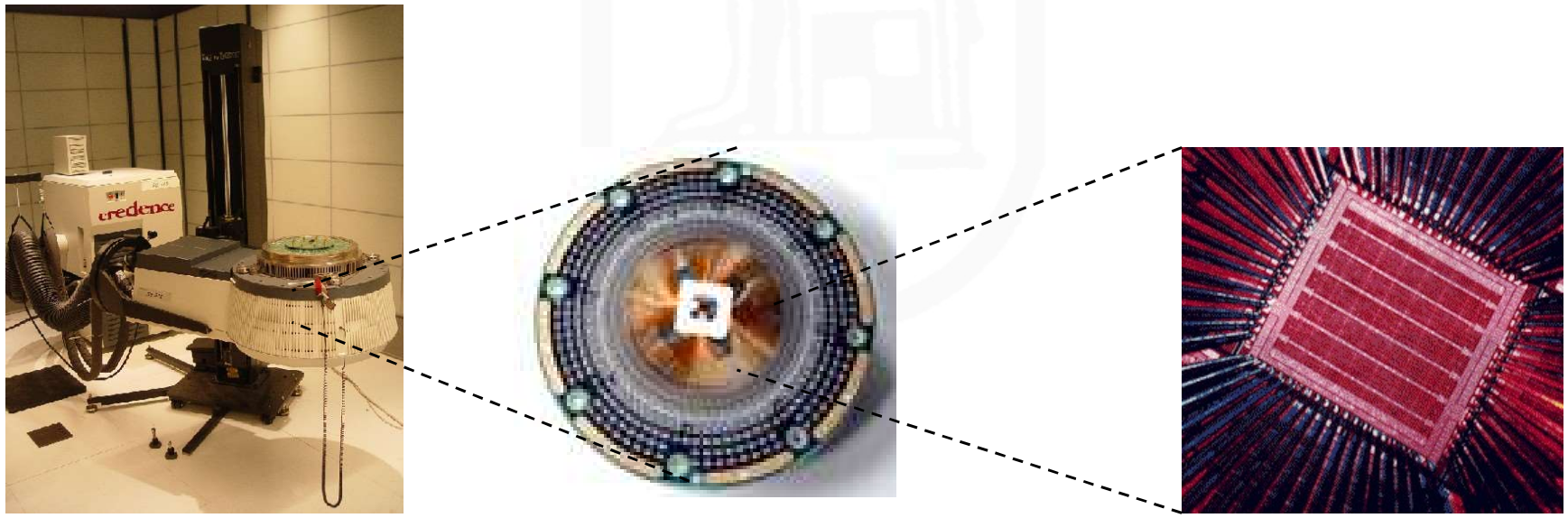
- **The defect level provides an indication of the overall quality of the testing process.**
- **Fault coverage is due to DFT scan and other techniques.**
- **A defect level of**
  - 500 parts per million (PPM) may be considered to be acceptable, whereas
  - 100 PPM or lower represents high quality.
  - The goal of six sigma manufacturing, also referred to as zero defects, is 3.4 PPM or less.

Example

- **Assume the process yield is 50% and the fault coverage for a device is 90% for the given test sets.**
  - By the preceding equation, we obtain $DL = 1 - 0.5^{1-0.9} = 0.067$
  - This means that 6.7% of shipped parts will be defective or the defect level of the product is 67,000 PPM.
- **On the other hand, if a *DL of 100 PPM is required for the same* process yield of 50%, then**
  - the fault coverage required to achieve the PPM level is $FC = 1 - (\log(1 - DL)/\log(Y) = 0.99986.$
  - it could be extremely difficult, if not impossible, to generate tests that have 99.986% fault coverage
  - Improvements of process yield might become mandatory in order to meet the stringent PPM goal.

IC Testing

- **Chip test needed after manufacturing so as to determine which parts are free of manufacturing defects**
  - **Goal is to discover manufacturing defects, not design bugs**
  - **Done before or after wafer dicing**



Wafer Sort

- Immediately after wafers are fabricated, they undergo preliminary tests in wafer sort.
- The tests applied at wafer sort may be a subset of the tests applied at final test after the chips are packaged.
- If a die fails a wafer sort test, it is marked with a drop of ink where only unmarked dies are packaged.
- Yield is determined as the fraction of the dies that pass the tests although the yield is different than the fabrication yield which is the percentage of wafers that make it through the entire fabrication process.

IC Testing

- **Because process yield and fault coverage affect defect level, reliability screen methods, are often used to accelerate the failure rate.**
  - Stress testing (through Burn-in) and
  - IDDQ testing (measuring leakage currents)
- **These methods weed out weak devices before mass production so as to reduce test escapes that will cause field returns.**
- **Once weak devices are found, FMA is performed to analyze, debug, locate, and correct the failures so the process yield can be increased later.**

IC Testing

- **Manufactured electronic systems shipped to the field may also undergo testing as part of the installation process to ensure a fault-free system before operation.**
- **During system operation, a number of events can result in a system failure;**
  - including single-event upsets (SEUs), electromigration, and material aging.
- **When an SoC design fails as a chip, on a board, or in the system, the ability to find the root cause of the failure in a timely manner becomes critical.**
- **Several IEEE standards (including 1149.1, 1149.4,**
- **1149.6, and 1500) and other techniques ease silicon test and debug as well as system-level test and diagnosis.**

Test Procedures

- **Reliability test**
  - Stress, IDDQ, etc.
- **Diagnostic test**
  - used in debugging and defect localization
  - can afford to spend time testing
- **Production test - "go/no go"**
  - used in chip production (wafer and/or packaged)
  - since have to test each part, must be fast: JTAG, DFT, etc.
- **Parametric test**
  - [v, i] versus [0,1]
  - check parameters such as noise margins, V_t, t_p at corners (range of temperatures and supply voltage variations)
  - usually done with special wafer drop-ins

Testing Fabricated Designs
Design-for-Test (DFT)

- **Goals of DFT**
  - make testing of manufactured parts swift and comprehensive
- **DFT mantra**
  - Provide controllability and observability
- **Components of DFT strategy**
  - Provide test patterns that guarantee reasonable coverage
  - Provide circuitry to enable testing

Two Important Test Properties

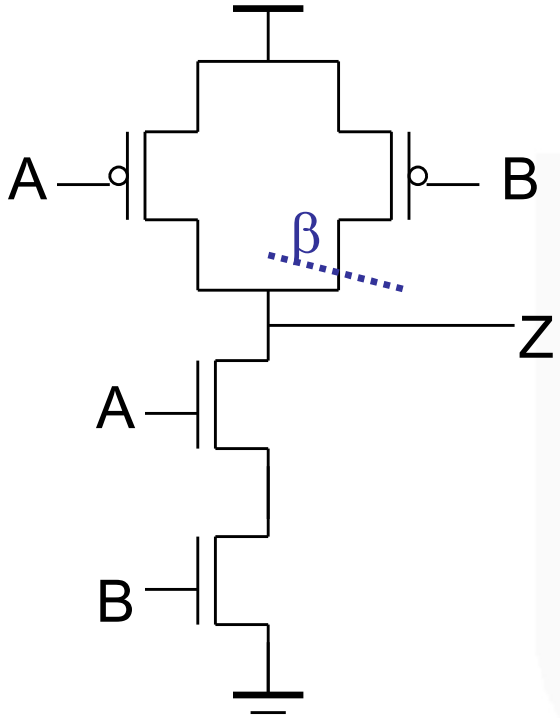
- **Controllability - measures the ease of bringing a node to a given condition using only the input pins**
- **Observability - measures the ease of observing the value of a node at the output pins**
- **Need both!**
  - combinational circuits are both - so relatively easy to determine test patterns
  - state in sequential circuits problematic - so turn into a combinational circuit (or use self-test)

Generating and Validating Test Vectors

- **Automatic test-pattern generation (ATPG)**
  - for given fault, determine excitation vector (called test vector) that will propagate error to primary (observable) output
  - majority of available tools: combinational networks only
  - sequential ATPG available from academic research
- **Fault simulation**
  - determines test coverage of proposed test-vector set
  - simulates correct network in parallel with faulty networks

- **Both require adequate models of faults in CMOS integrated circuits**

Problem with Stuck-at Model



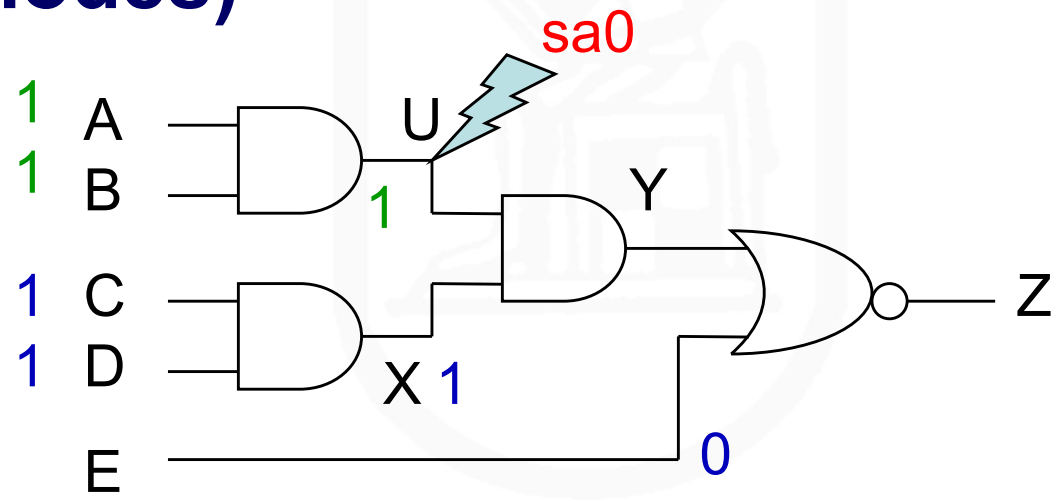
A	B	Z
1	1	0
0	-	1
1	0	Z_{i-1}

output node floats
(retains old value)

sequential effect - needs two vectors to ensure detection

Path Sensitization

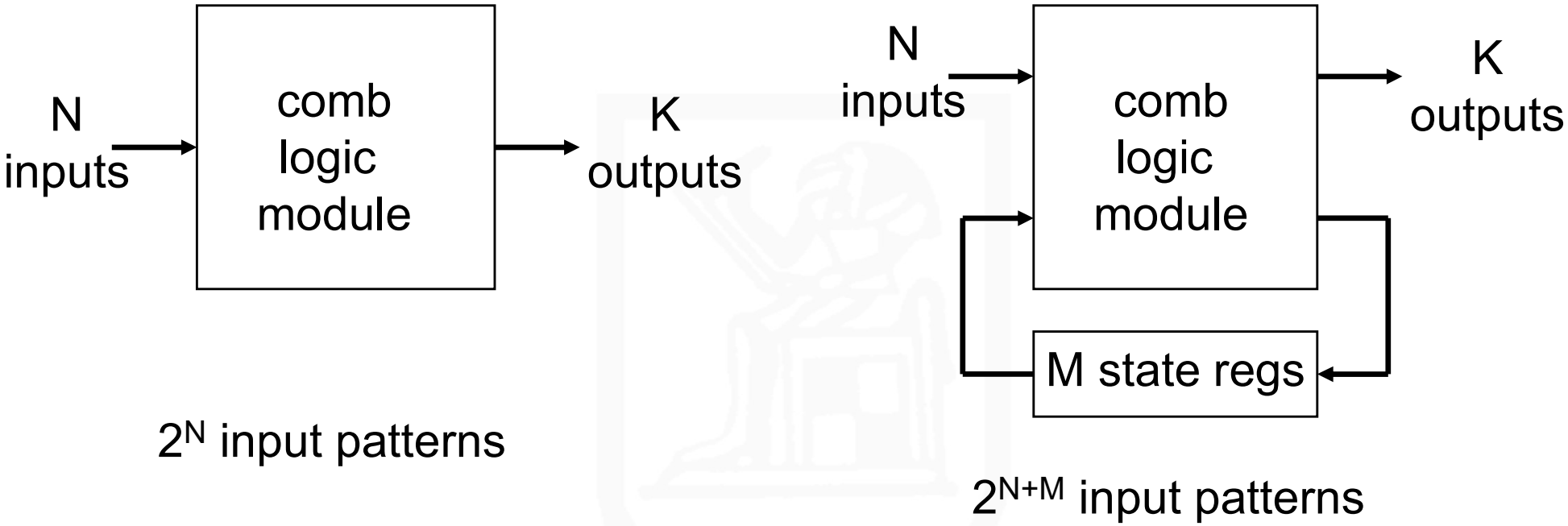
- **Determine the input pattern that makes a fault controllable (triggers the fault) and observable (makes its impact visible at the output nodes)**



Controllable: Try to set U to 1 $\rightarrow$ A = 1 and B = 1

Observable: Propagate U to Z $\rightarrow$ X = 1 and E = 0

Test Problem Size



2^N input patterns

2^{N+M} input patterns

N=20 → 1 million patterns
1 μsec/pattern → 1 second test

N=20, M=10 → 1 billion patterns
1 μsec/pattern → 16 minute test

Reducing Number of Test Vectors

- Two features can be exploited to reduce the number of test vectors

- Redundancy - a single fault in the circuit is usually covered by several input patterns; detection of the fault requires only one
- Reduced fault coverage - relax the requirement that all faults be detected (95% to 99% fault coverage is typical)

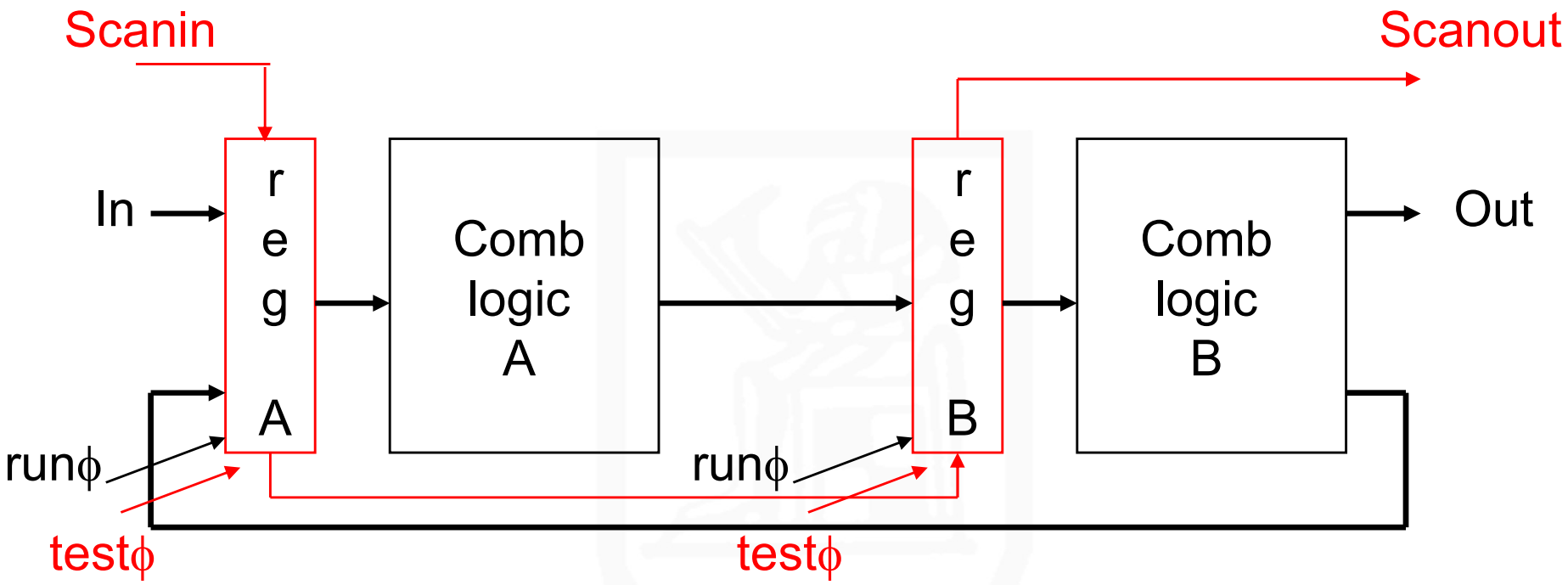
Test Approaches

- **Scan based test**
- **Self test**
- **Ad-hoc testing**

Problem is getting harder

- increasing complexity and heterogeneous combination of modules in system-on-a-chip.
- advanced packaging and assembly techniques extend problem to the board level

Scan Based Test

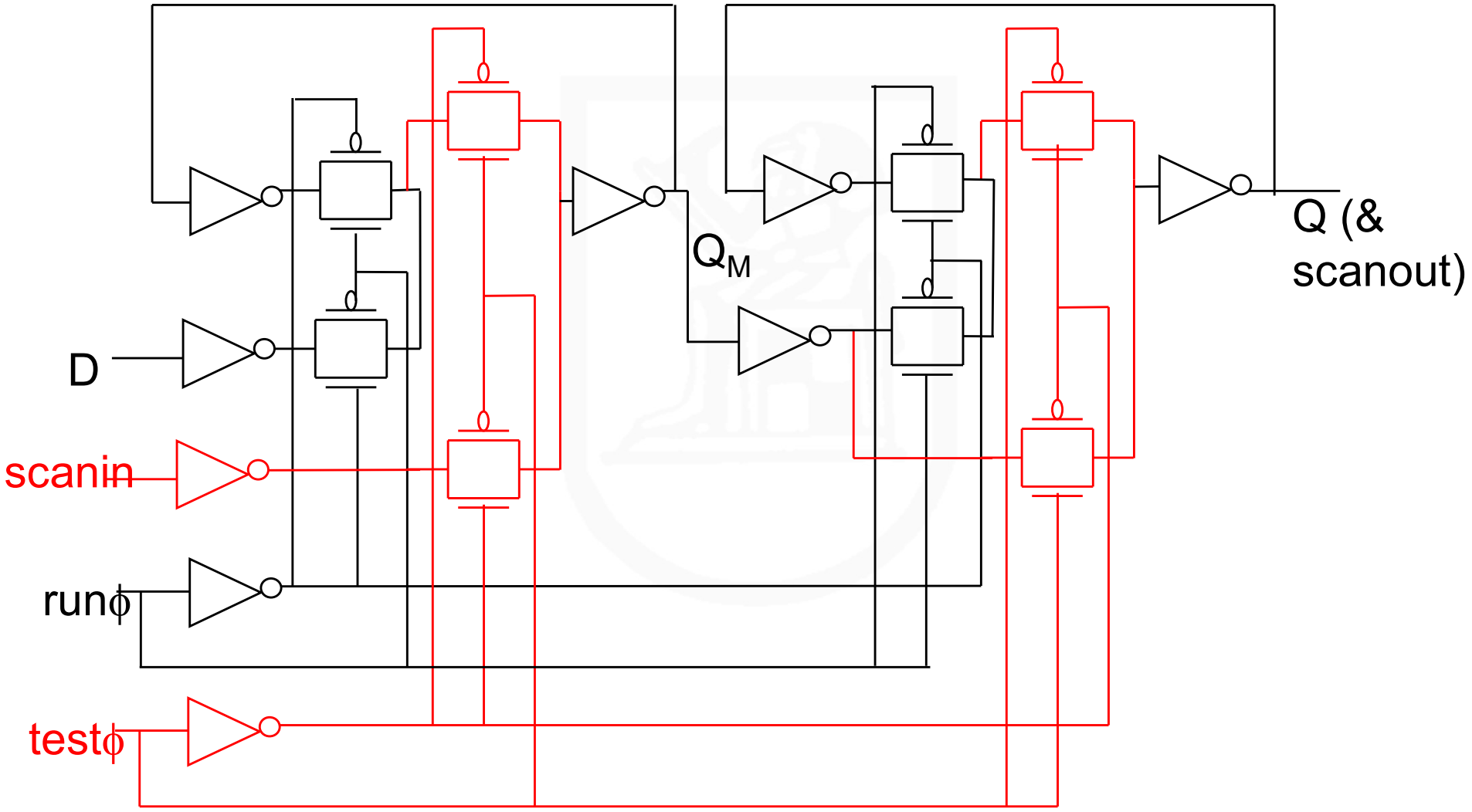


Two operational modes

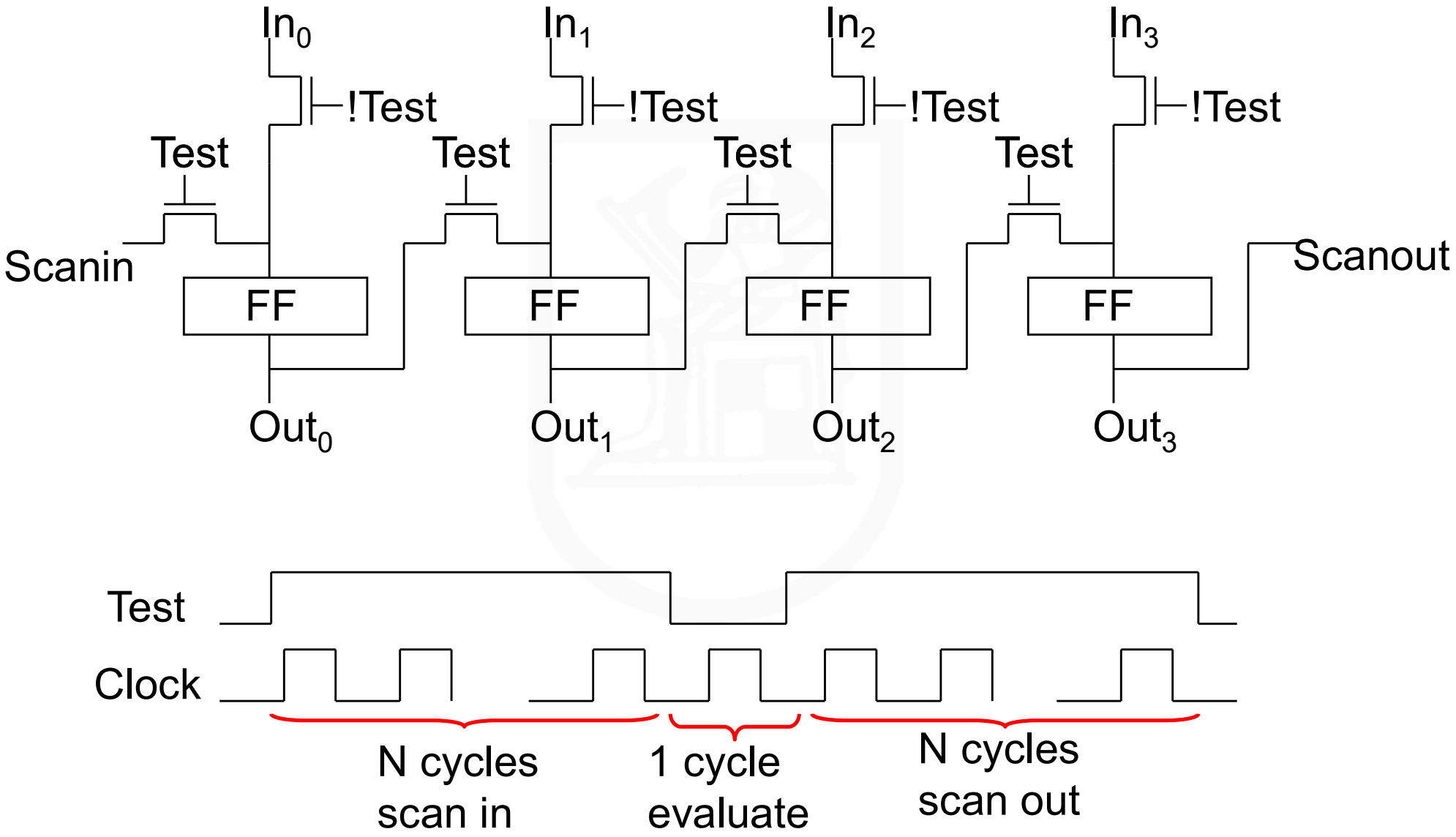
normal mode (N-bit wide clocked registers)

test mode (registers chained as a single serial shift register)

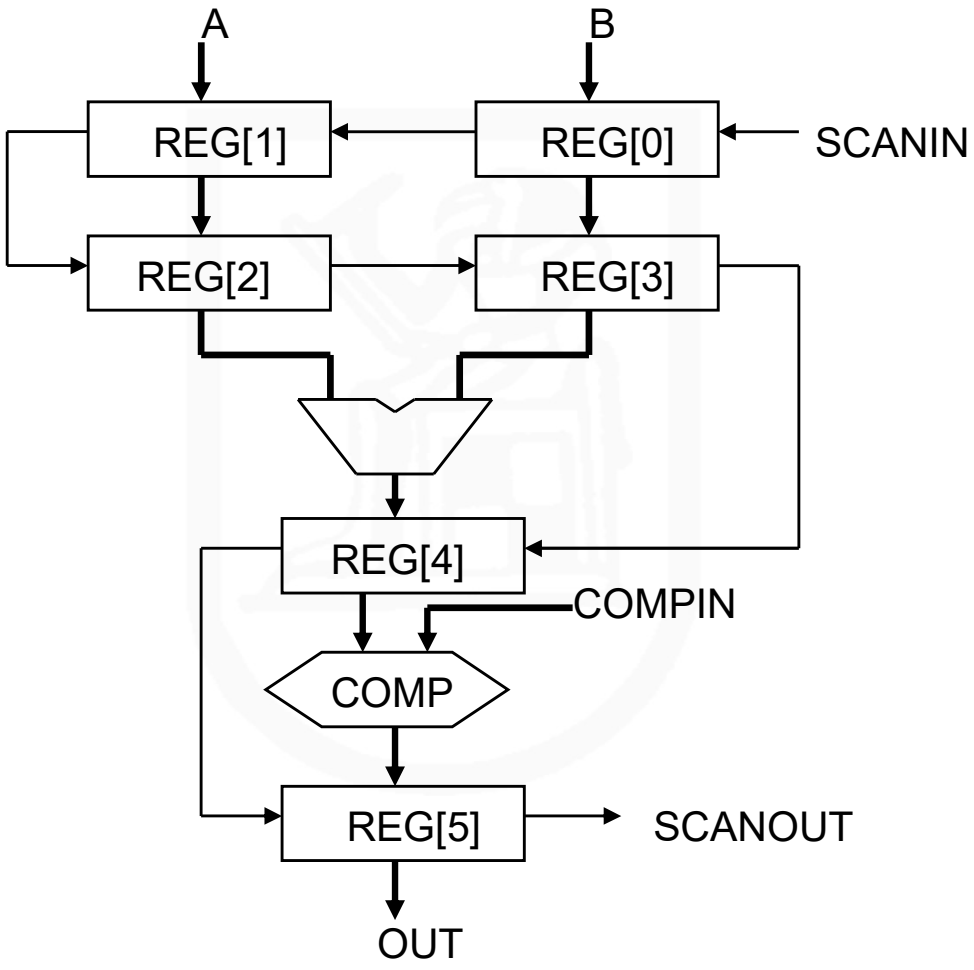
Scan Path FF Implementation



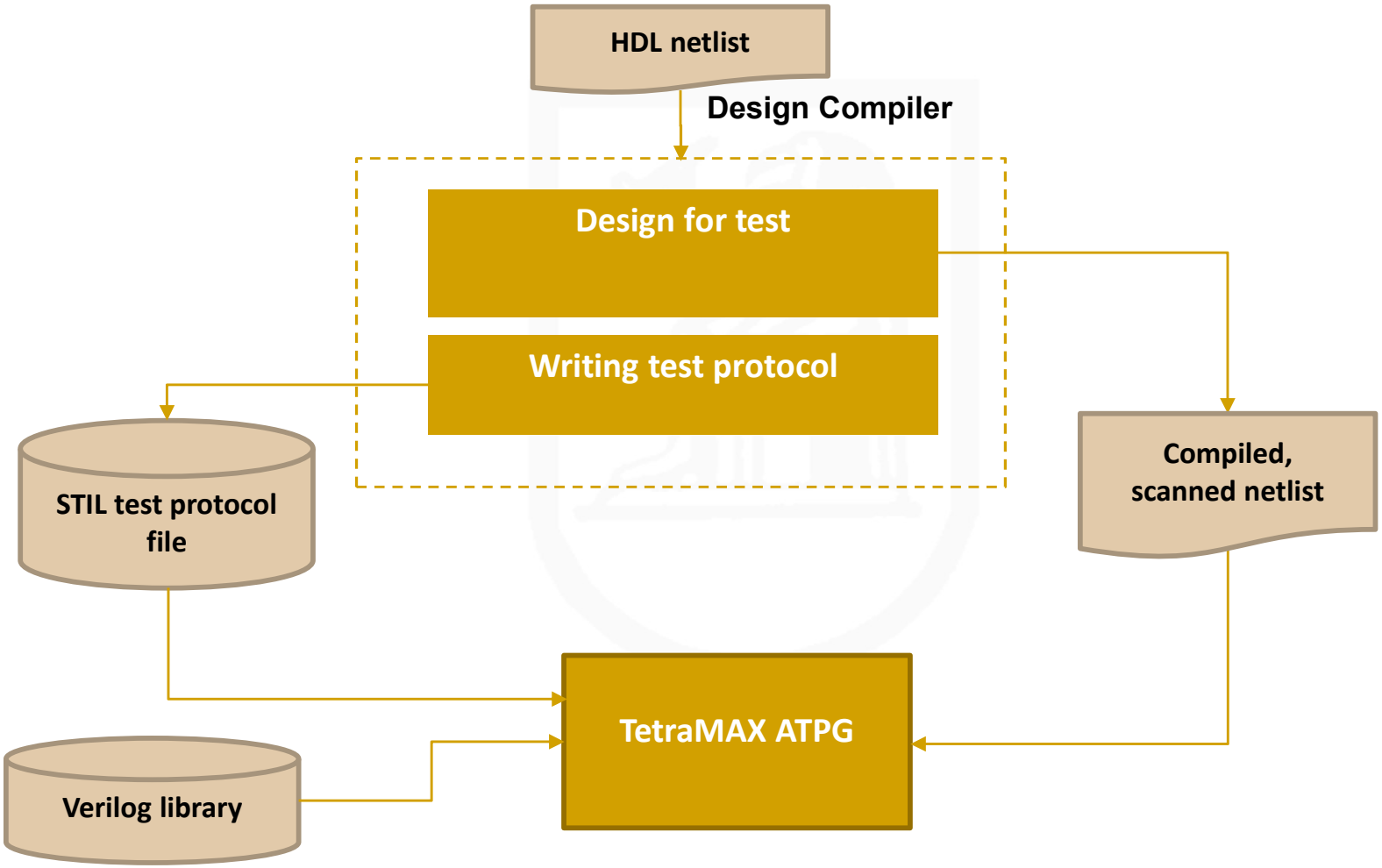
Scan Register



Scan Path Testing



Design Flow Using DFT Compiler and TetraMAX



Scan DFT in the Synopsys Environment

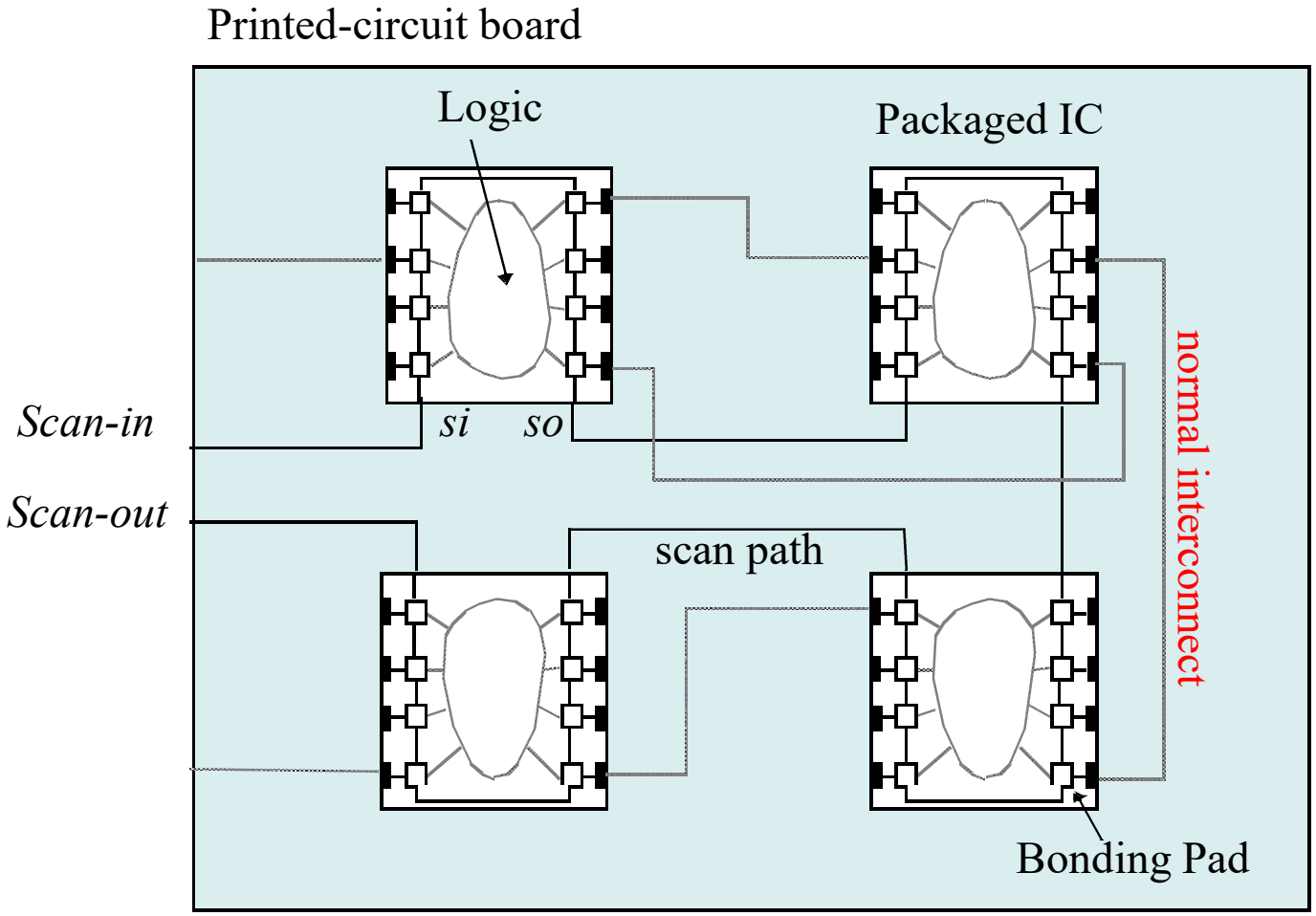
Sample Synopsys script with a concentration on DFT.

```
read -f Verilog fsm.v
link_library = target_library = lsi_10k.db
create_clock clock81 -period 12.3

/* must expand design and click on clock81 first */
set_input_delay -clock clock81 -max -rise 2 “RW”
set_test_methodology full_scan

/* menu: attributes -> optimize directives -> design */
set_scan_style multiplexed_flip_flop

/* compile including scan */
compile map_effort low
```

Scan DFT in the Synopsys Environment

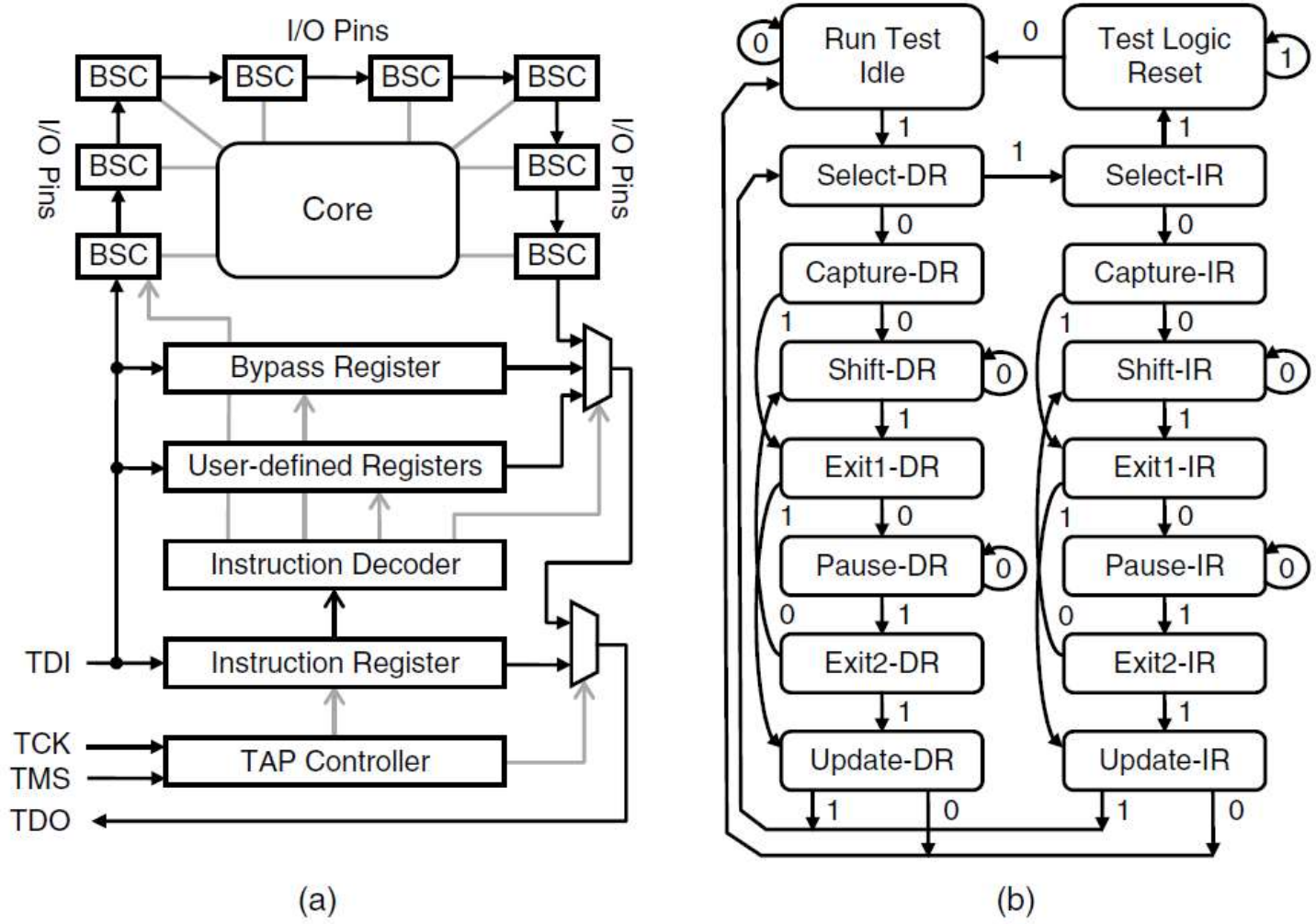
```
/* check for testability analysis -- look at result */
insert_test
check_test

/* create test patterns to check for ATPG conflicts (additional option in */
/* create pattern menu), also checks fault coverage */
create_test_patterns -sample 5

/* full test pattern creation and scan insertion are done complete chip at */
/* the end but try them if you want */
insert_test -scan_chains 1
create_test_patterns -output fsm.vdb \
-compaction_effort low \
-check_contention_float true -backtrack_effort low \
-random_pattern_failure_limit 64 -sample 100
```

Boundary Scan (JTAG)



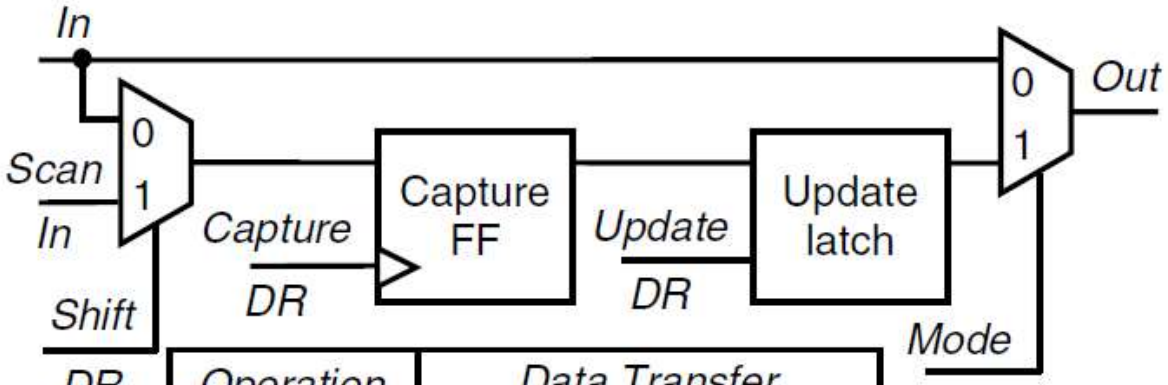
Board testing becoming as problematic as chip testing

Boundary Scan or JTAG (IEEE 1149.1 Standard)



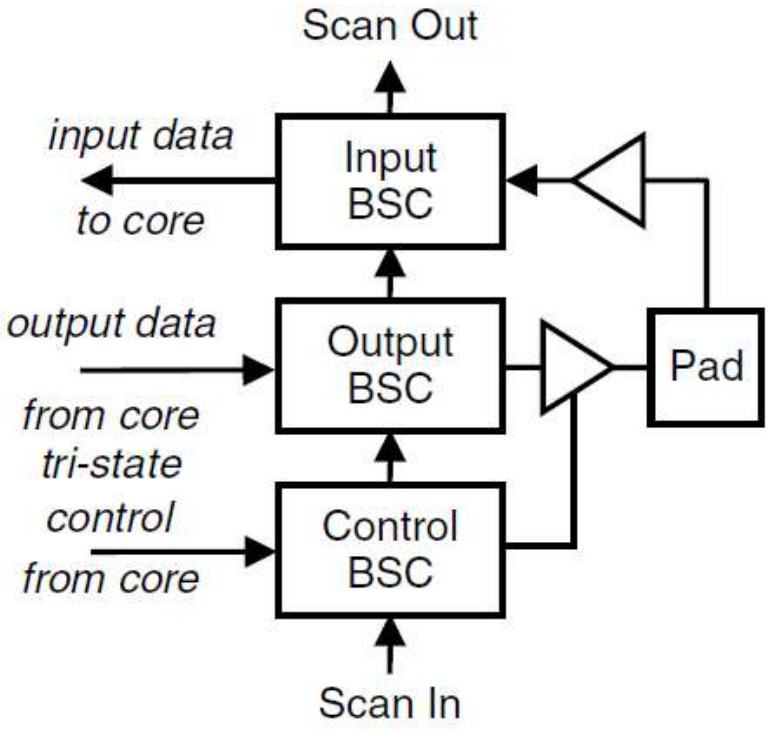
Boundary-scan interface: (a) boundary-scan implementation and (b) TAP controller state diagram.

Boundary Scan Cell



Operation	Data Transfer
Normal	Out ⇐ In
Scan	Capture ⇐ Scan In
Capture	Capture ⇐ In
Update	Update ⇐ Capture

(a)



(b)

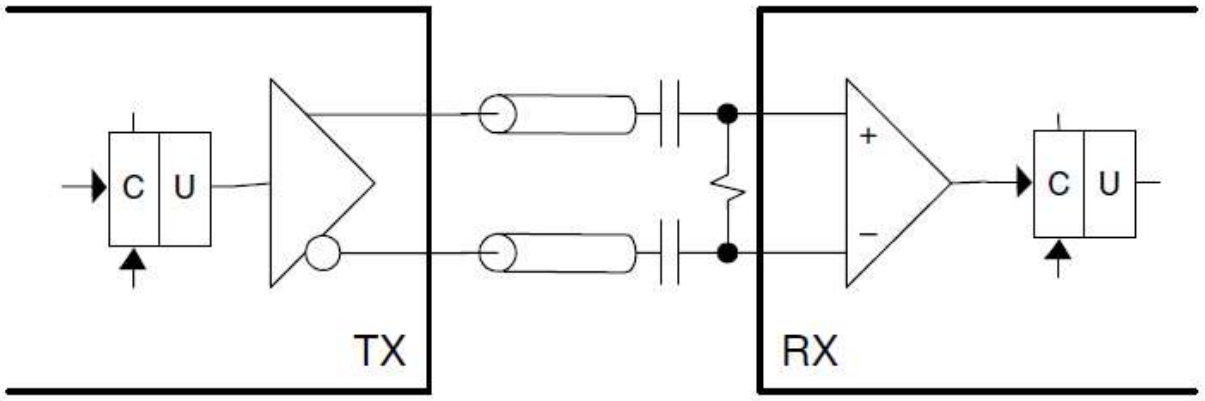
Boundary-scan cells: (a) boundary-scan cell (BSC) and operation modes and (b) bidirectional buffer with BSCs.

Significance of 1149.1 Boundary Scan

- **Provides an approach to testing interconnects on digital PCBs.**
- **Overcomes the test problems encountered with surface mount technology emerging in the late 1980s.**
- **Does not directly address test logic internal to the devices on a PCB but**
  - it does provide a standardized interface that can be used to access internal test mechanisms, such as scan chains and BIST circuits, designed specifically for the internal logic testing.
- **More importantly, it provides a proven solution to the test problems that would later be encountered with SoC and SIP implementations.**

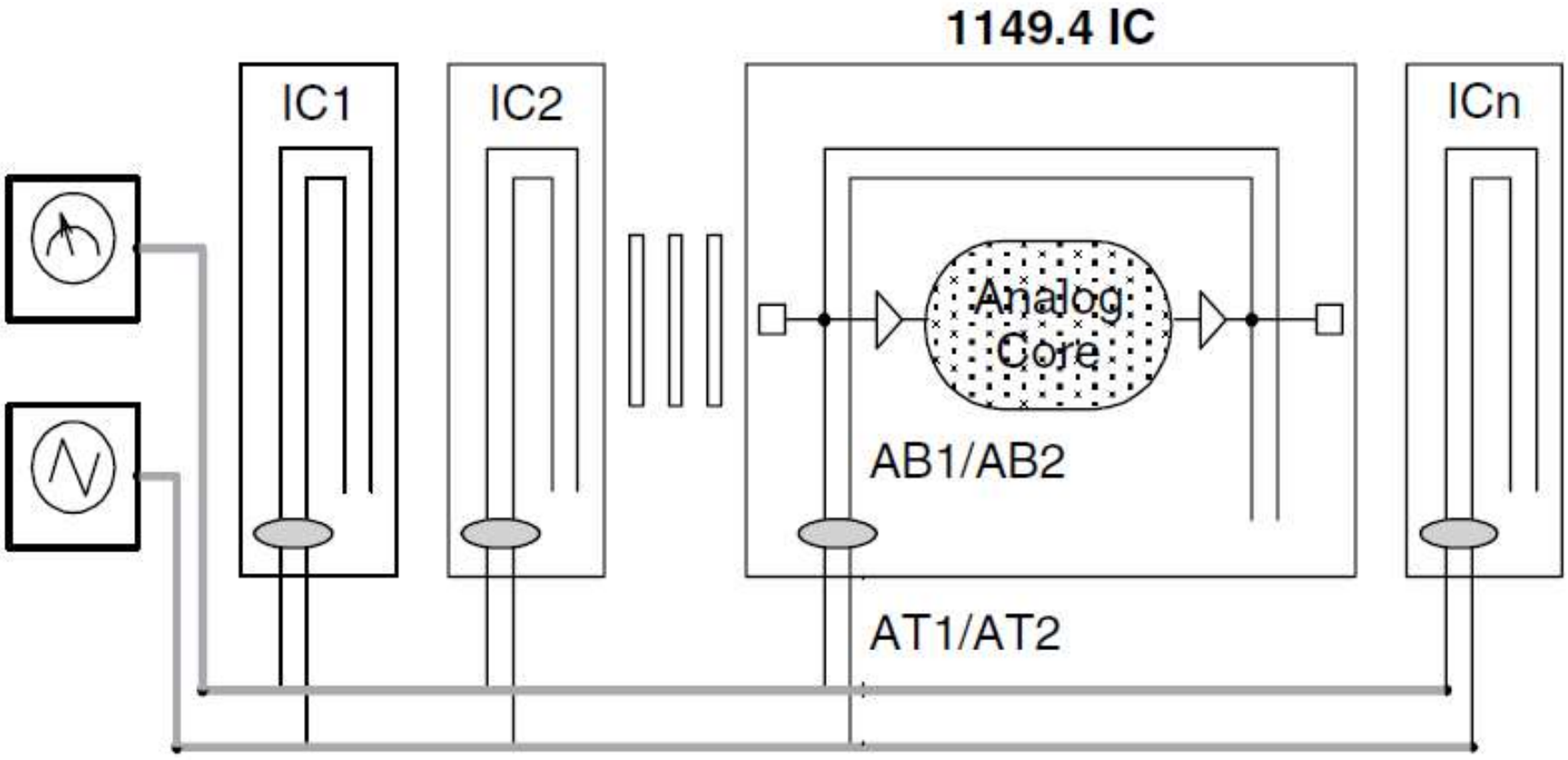
Boundary Scan Extension (IEEE 1149.6 Standard)

- **IEEE 1149.6 standard is an extension of the IEEE 1149.1 standard;**
- **Must comply with all 1149.1 rules.**
- **Allows for testing of high-speed AC-coupled digital interconnects by**
  - capturing edges of pulses that are generated by 1149.6 drivers.
  - A special, analog test receiver is used to capture these edges.
  - The 1149.6 receiver logic is placed on both inputs of the differential receiver logic.
  - Special hysteresis logic filters out noise and captures only valid transitions.



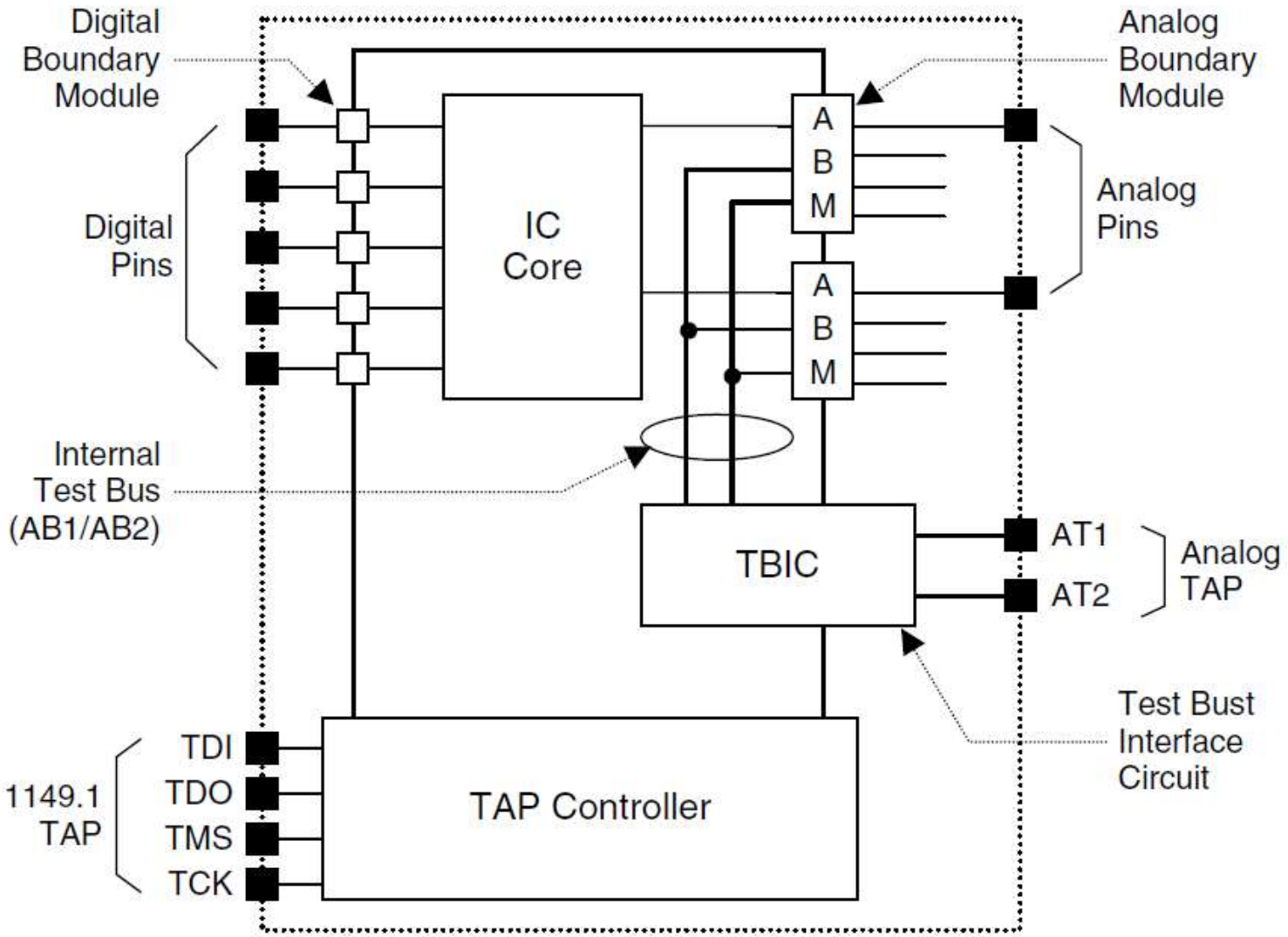
Differential, AC-coupled network

Analog Boundary Scan (IEEE 1149.4 Standard)

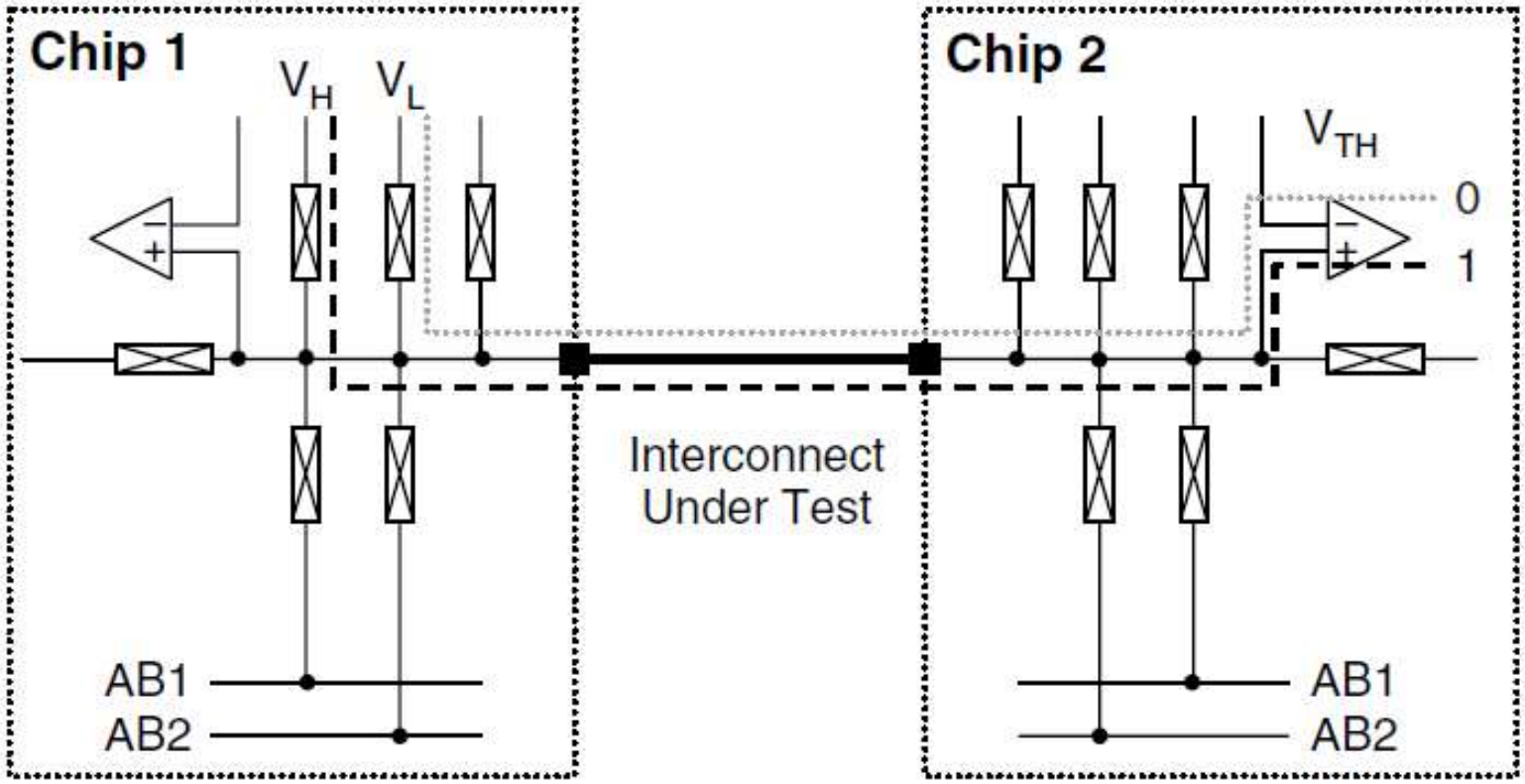


IEEE 1149.4 internal test configuration

IEEE 1149.4 Compliant Chip Architecture

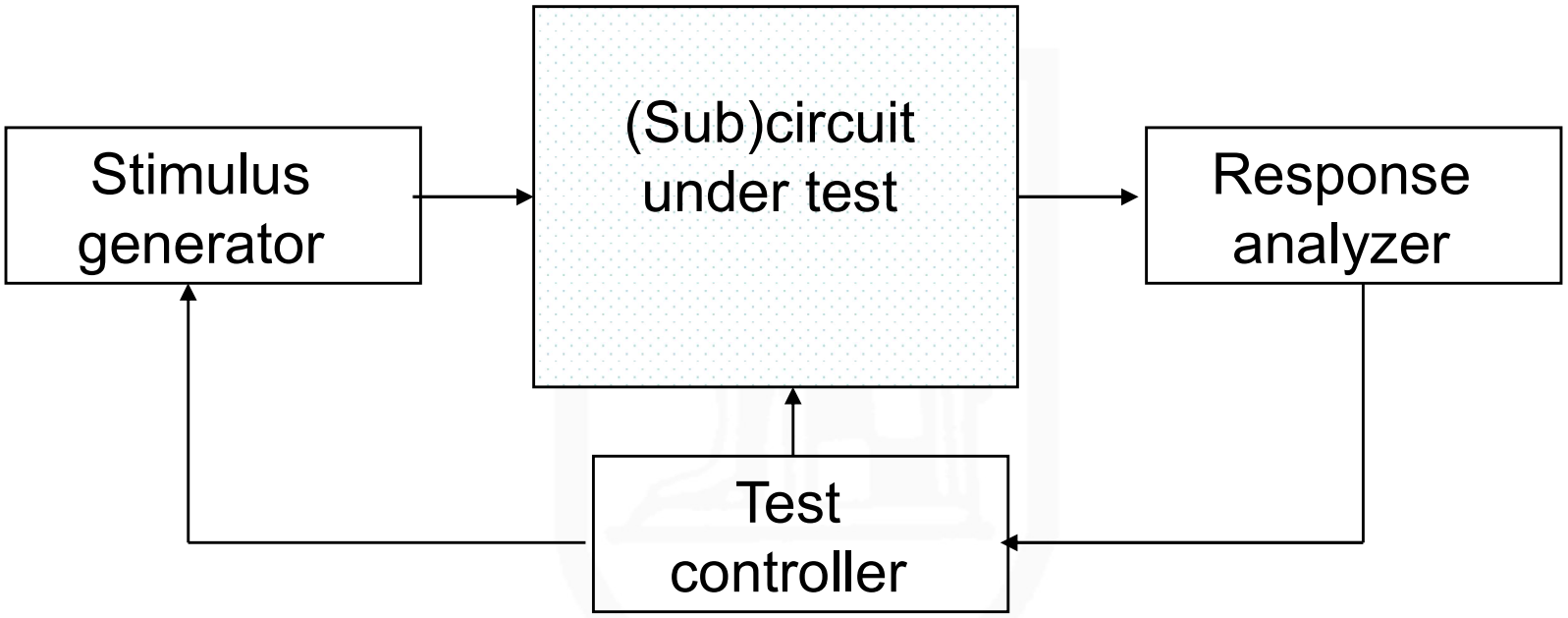


Interconnect Testing Example



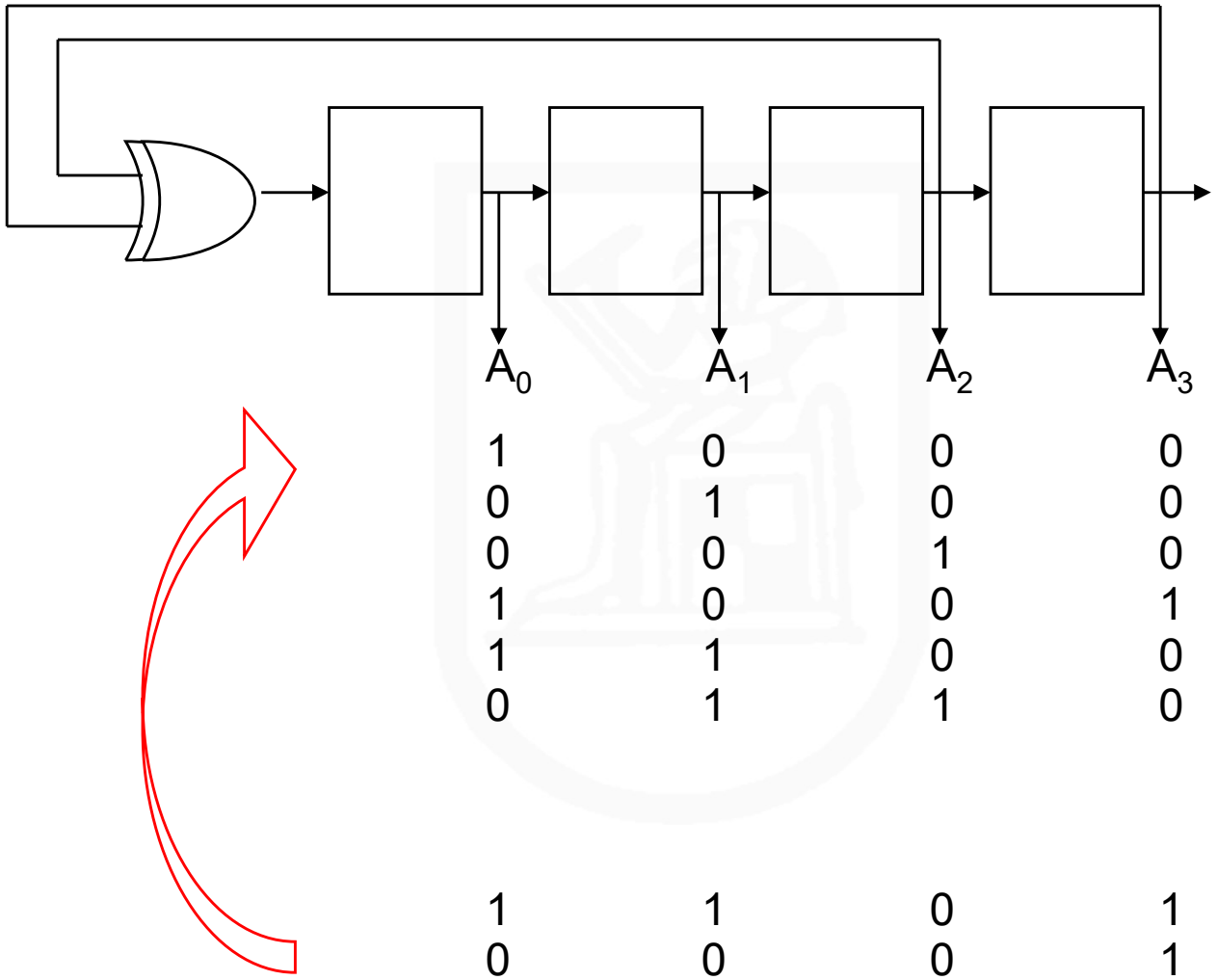
Built in Self Test (BIST)

- **The circuit decides if the results are correct!**



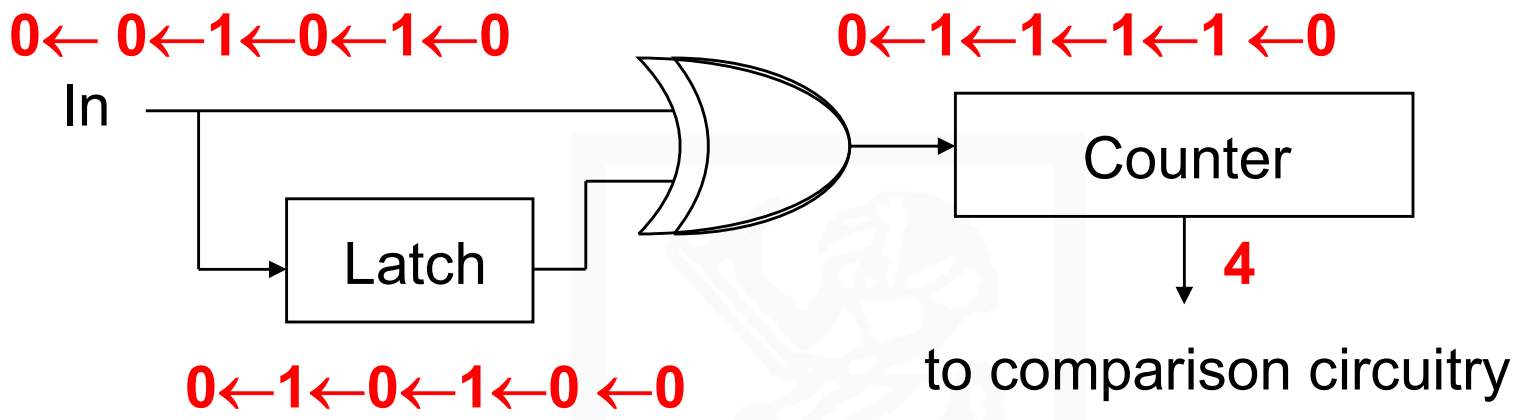
- **Need a way to supply test patterns (stimulus generator) and to compare the circuit's responses to a known correct sequence (response analyzer)**

Stimulus Generator (LFSR)



Pseudo-random pattern generator

Response Analyzer

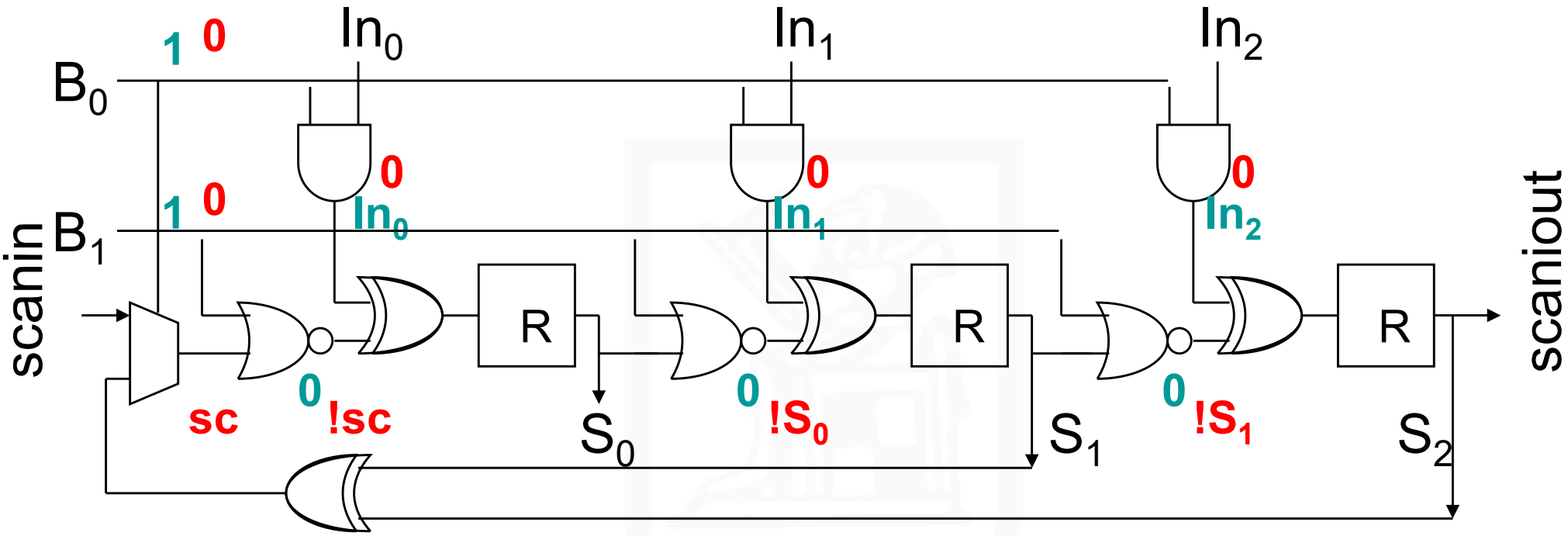


Counts the number of 0 → 1 and 1 → 0 transitions

Counter value (signature of the circuit) is then compared to the known correct count

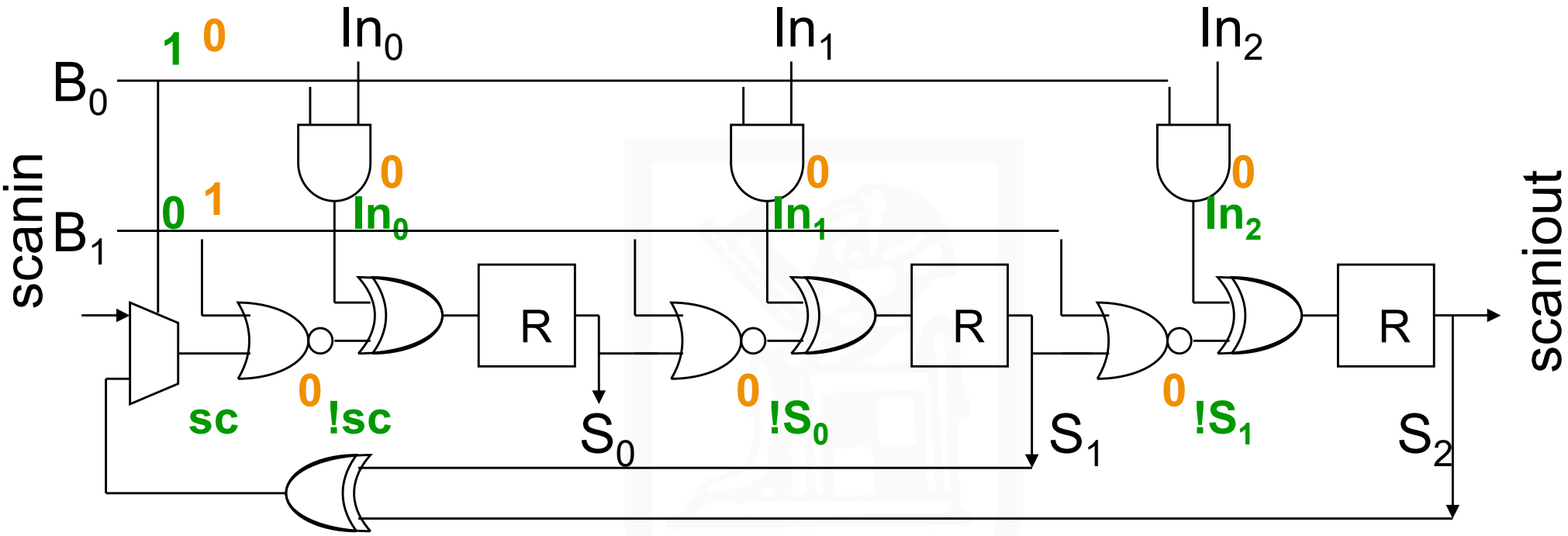
Signature Analysis

BILBO



B_0	B_1	Op Mode
1	1	Normal
0	0	Scan
1	0	
0	1	

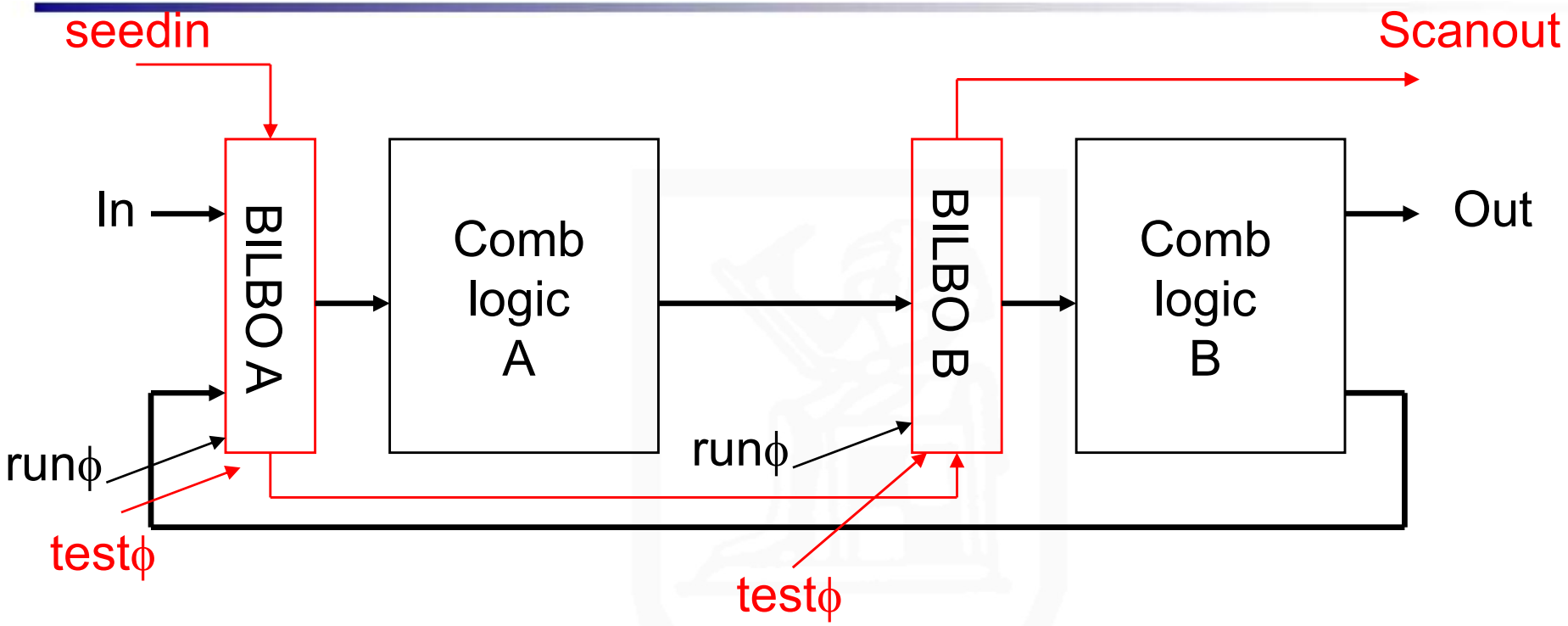
BILBO



B_0	B_1	Op Mode
1	1	Normal
0	0	Scan
1	0	BIST
0	1	Reset

$In_i \oplus !S_{i-1}$

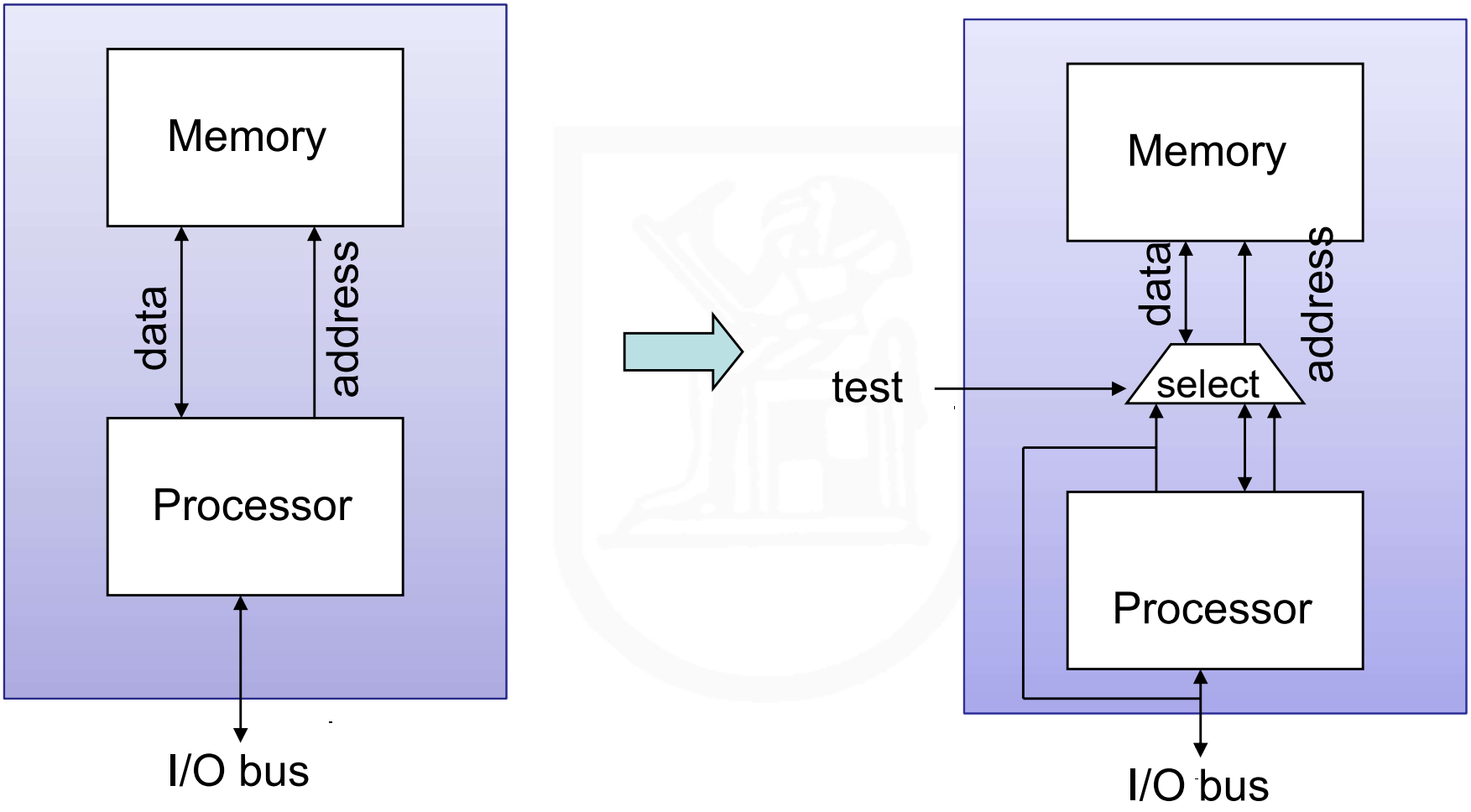
Use of BILBO



Reg A is a BILBO register doing pattern generation to test Comb logic A

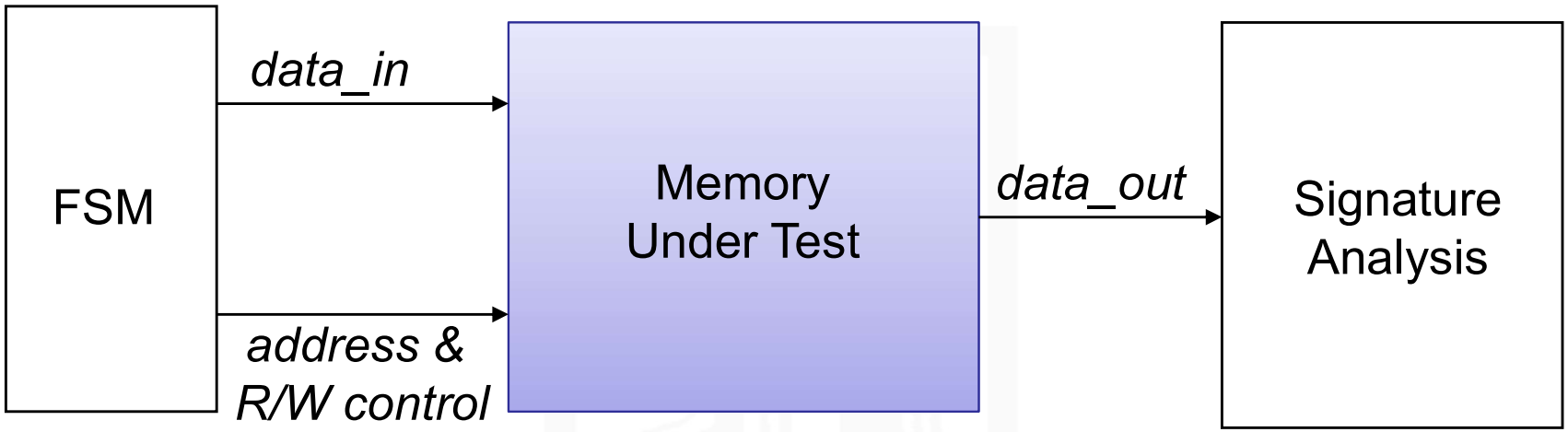
Reg B is a BILBO register doing signature analysis to test Comb logic A

Ad-Hoc Test



Inserting multiplexer improves testability

Memory Self Test



Patterns: Writing/Reading 0s, 1s,
Walking 0s, 1s
Galloping 0s, 1s