INTERNATIONAL ATOMIC ENERGY AGENCY
UNITED NATIONS EDUCATIONAL, SCIENTIFIC AND CULTURAL ORGANIZATION
INTERNATIONAL CENTRE FOR THEORETICAL PHYSICS
I.C.T.P., P.O. BOX 586, 34100 TRIESTE, ITALY, CABLE CENTRATOM TRIESTE

The United Nations University

SMR/748 - 3

ICTP-INFN-UNU-MICROPROCESSOR LABORATORY
THIRD COURSE ON BASIC VLSI DESIGN TECHNIQUES
2 November - 16 December 1994

DESIGN FOR TESTABILITY
(+ Cheap PC CAE tool)

Jørgen CHRISTIANSEN
ECP Division
CERN
1211 Geneva 23
Switzerland

Design for testability

(+ Cheap PC CAE tool)

Jørgen Christiansen CERN / ECP, 1211 Geneve 23, Switzerland
(TEL: +41 22 767 5824 , Email:CHRISTIA@SUNVLSI.CERN.CH)

Overview

Basic testing theory:

- Why testing: cost of testing and yield.
- Reliability of VLSI circuits.
- What to test : Combinatorial, Sequential, Memory.
- Basic testing terms, fault models.
- Fault coverage.
- Generation of test patterns.
- Memory testing.
- Steady state power supply current testing.
- VLSI testers.
- E-beam testing.
- Test of analog IC's.

Design verification testing:

- Chips working in final application.
- In circuit emulation.

Scan path testing:

- Improving controllability and observability using scan paths.
- JTAG (Joint test action group), IEEE standard 1149.1.
- JTAG protocol.
- Boundary test.
- Typical JTAG scan path cells.
- JTAG ASIC libraries.
- JTAG test equipment.
- Alternative use of JTAG.

Built in self test (BIST):

- Different schemes of BIST.
- Pseudo random generators.
- Signature analyzing.
- Built in logic block observer (BILBO).
- Running self test via JTAG.

Design for testability guidelines.

Testing seen by an ASIC designer.

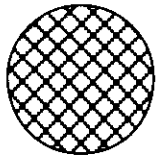
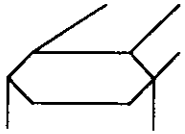
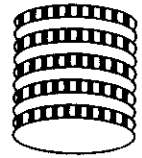
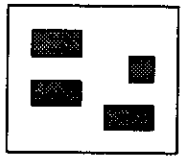
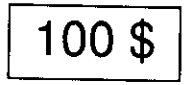
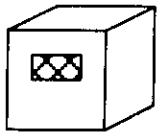
A cheap and easy to use PC chip design tool set:

- Functions in high end chip design tools.
- Required data from chip vendor.
- Example of small cheap PC based design system.
- Public domain software (MAGIC).

Basic testing theory

Why testing

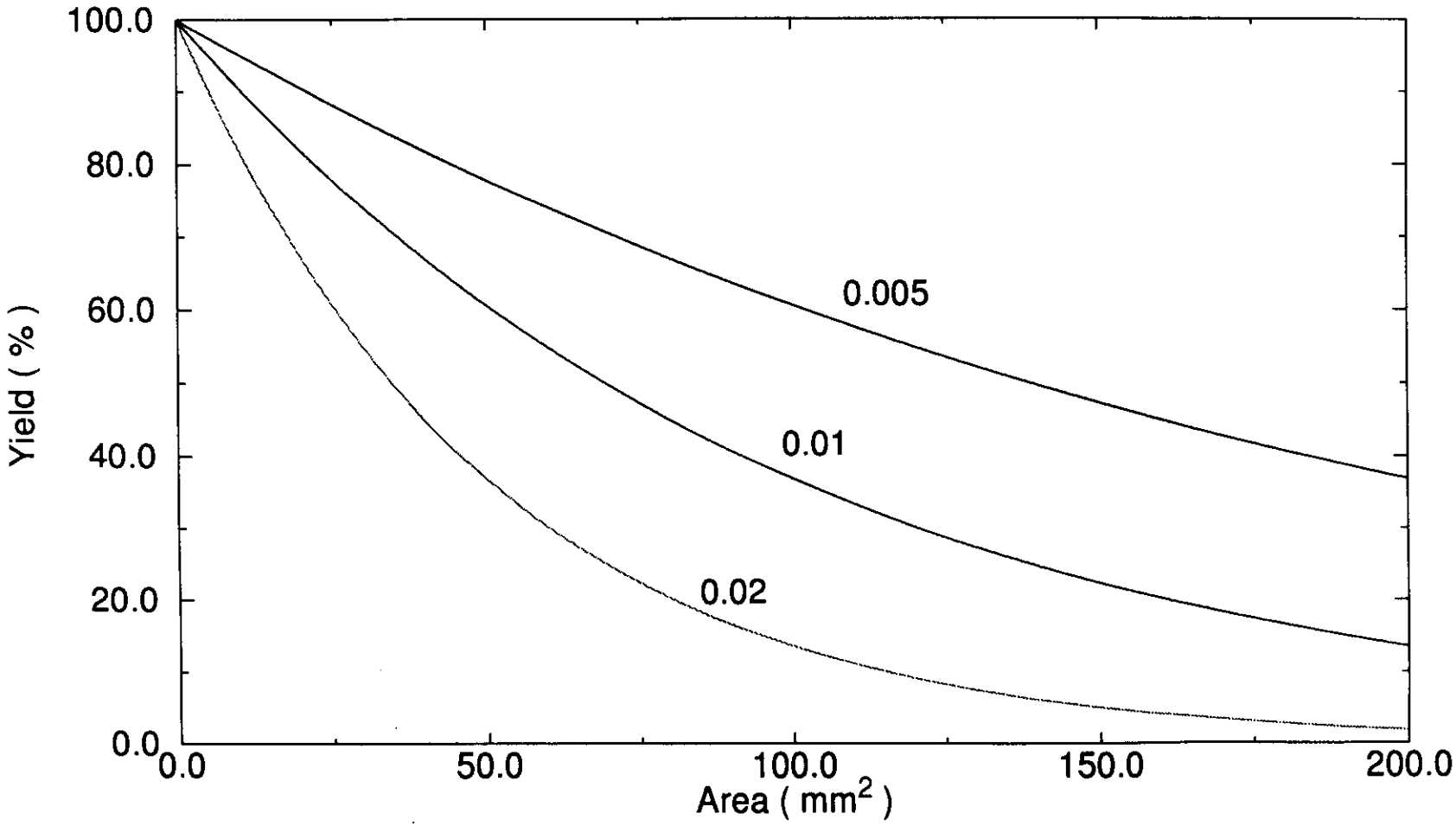
Price of finding and repairing a failing component

LEVEL	FAILURE MECHANISM	PRICE
Wafer 	Chip process: shorts, diffusion, etc.	1$ 
Chip 	Cutting, bonding	10$ 
Module 	Soldering, bended pins, ESD	100$ 
System 	Cables, connectors	1000$ 
At user	Reliability of components, vibrations, corrosion	10.000$ GOLD

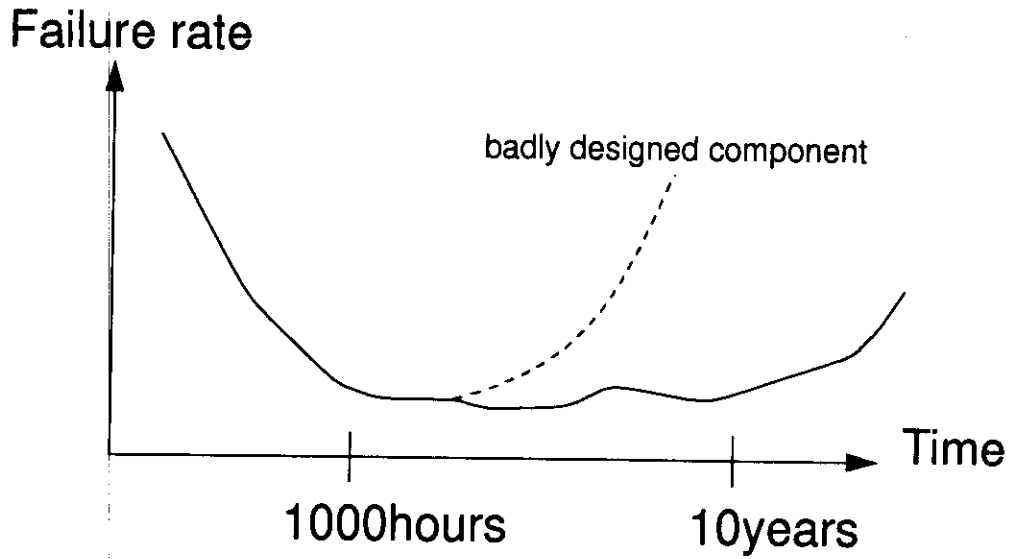
Yield

Yield is calculated from defects per mm^2 (= exp(- A * D))

Typical defect density is of the order of 0.005 - 0.02 defects/mm^2



Reliability of VLSI circuits



Failing parts within first 1000 hours : 1 - 5 %

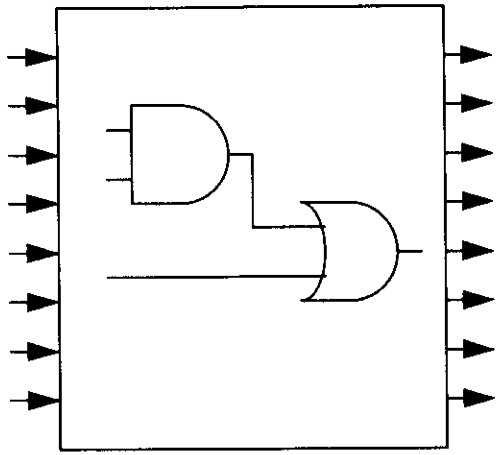
Burn-in testing : Heating up chips to 125 deg. accelerates 1000 hours period to approx. 24 hours.

Static: power supply connected.
Dynamic: Power + stimulation patterns.
Functional test: Power + stimulation patterns + test.

Temperature cycling: Continuous temperature cycling of chips to provoke temperature gradient induced faults.
(Non matching thermal expansion coefficients).

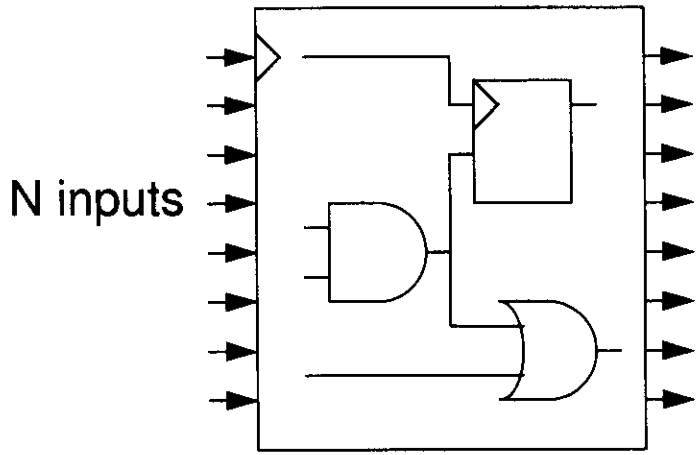
What to test

Combinatorial



Exhaustive test vectors: 2^N

Sequential



M storage elements

Exhaustive test vectors: $2^{(N+M)}$

10 Mhz tester:
N=32 ; test time = 7 min.
N=64 ; test time = 60.000 years

Knowledge about topology of circuit must be used to reduce number of test vectors so they can be generated by tester (tester memory: 10K - 10M).

Basic testing terms

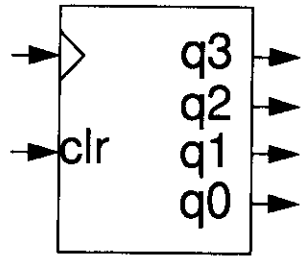
- CONTROLABILITY: The ease of controlling the state of a node in the circuit.
- OBSERVABILITY: The ease of observing the state of a node in the circuit

Example: 4 bit counter with clear

Control of q3:

Set low: perform clear = 1 vector

Set high : perform clear + count to 1000B = 9 vectors

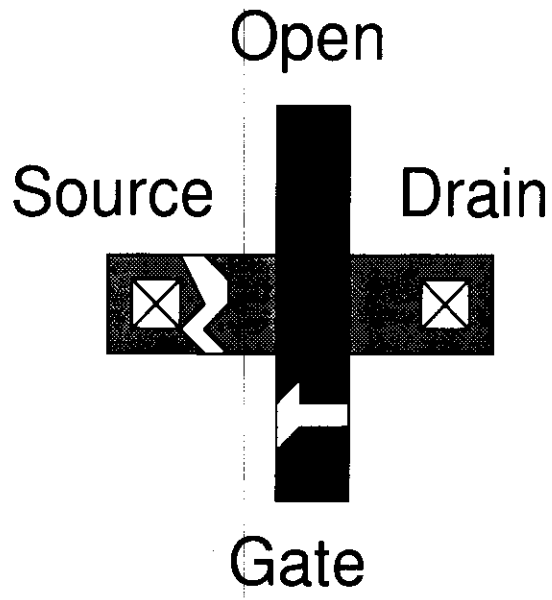


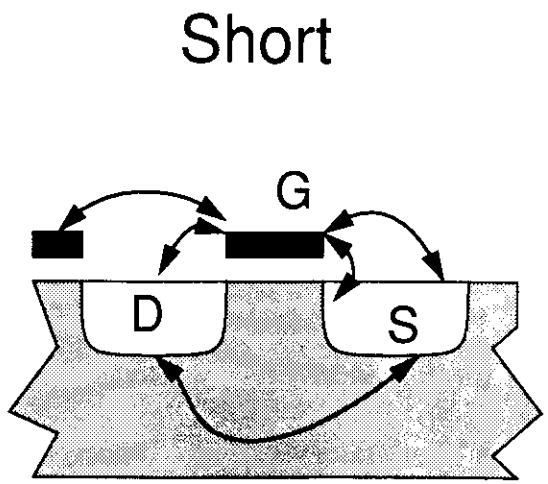
Testing a node in a circuit

- A: Apply sequence of test vectors to circuit which sets node to demanded state.
- B: Apply sequence of test vectors to circuit which enables state of node to be observed.
- C: The observing test vector sequence must not change state of node.

Fault models

- Fault types: Functional.
  Timing.
- Abstraction level: Transistor. (layout)
  Gate. (netlist)
  Macro (functional blocks).

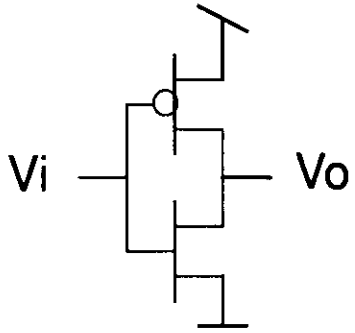


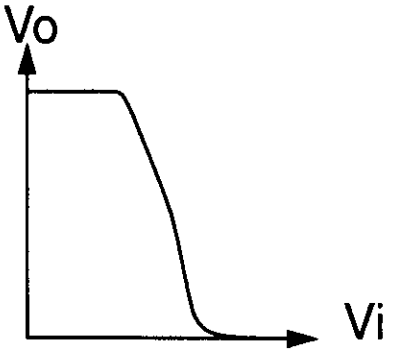


Transistor level

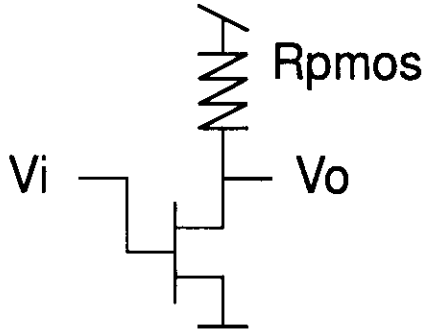
Transfer characteristic

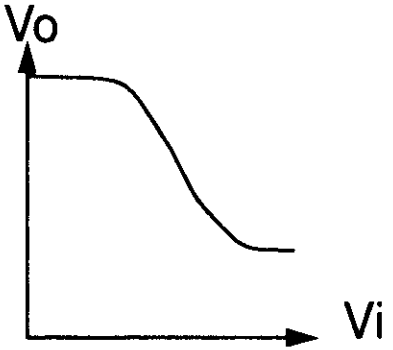
Full functional inverter





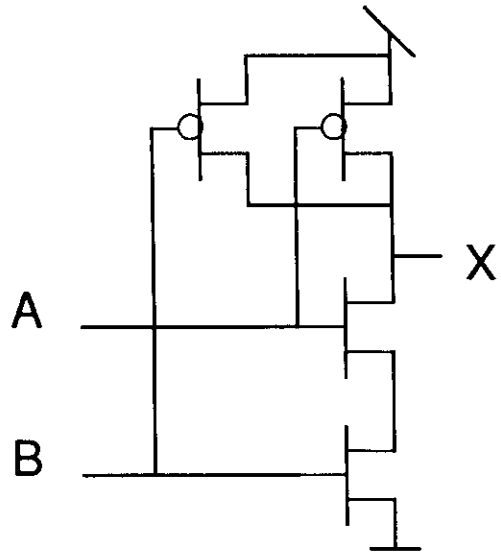
PMOS stuck on





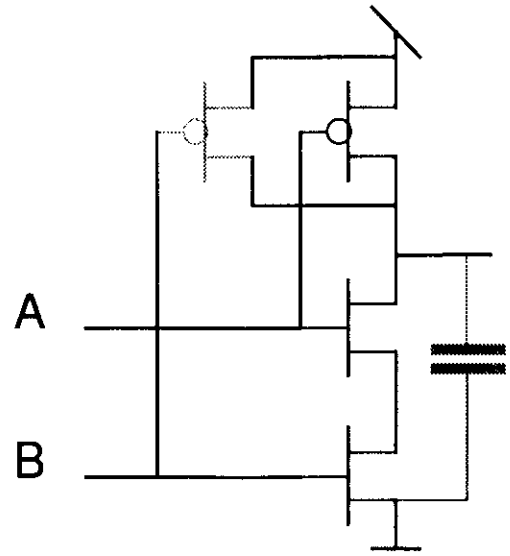
CMOS logic may become NMOS logic if PMOS transistor stuck on.

Full functional NAND



A	B	X
0	0	1
0	1	1
1	1	0
1	0	1

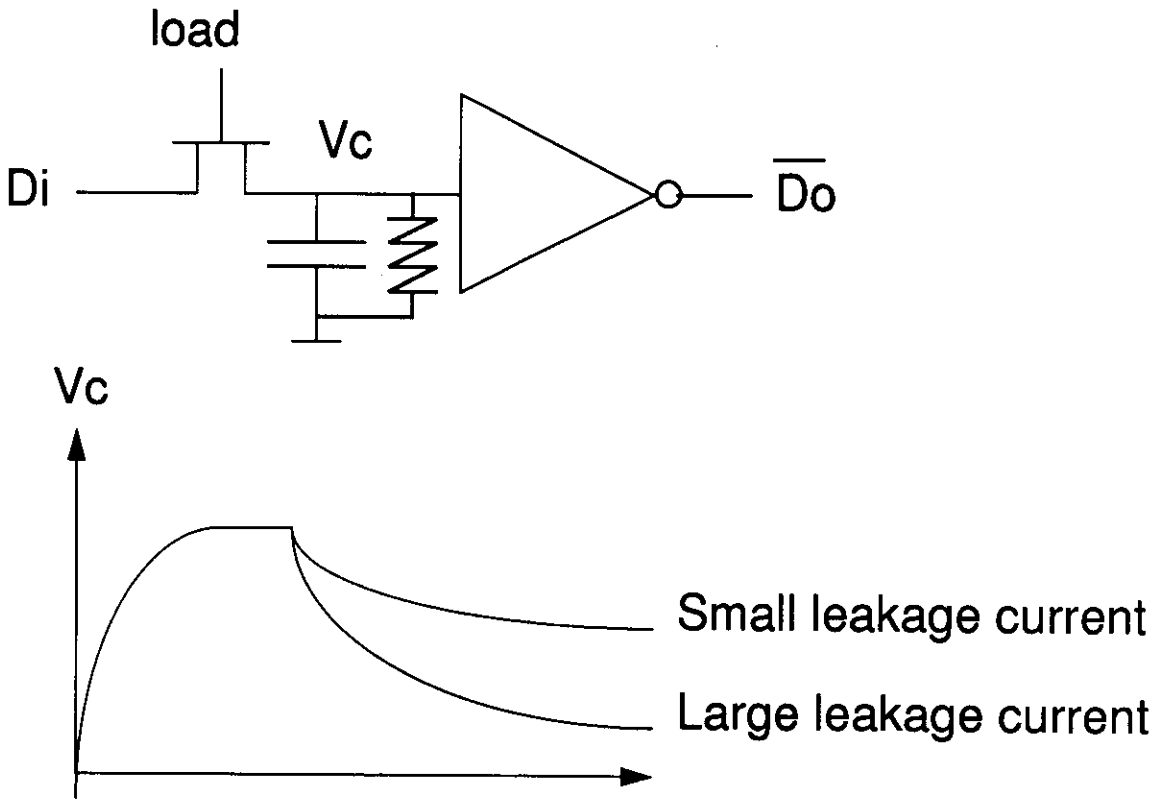
One PMOS stuck open



A	B	X
0	0	1
1	0	1
1	1	0
0	1	1

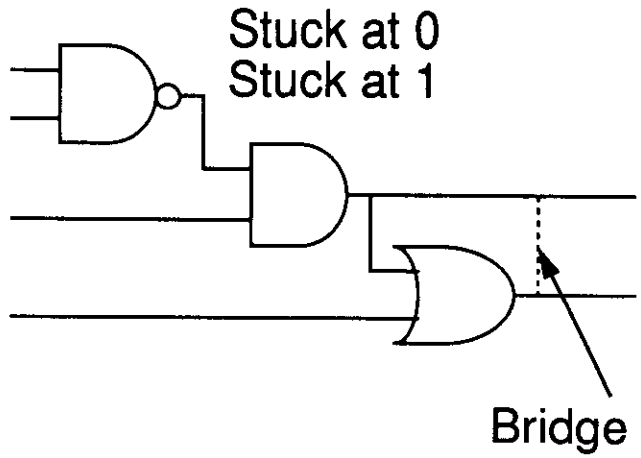
A	B	X
0	0	1
0	1	1
1	1	0
1	0	0

Combinatorial logic may become sequential if stuck open faults



Dynamic storage elements may loose information if circuit run at low frequency.

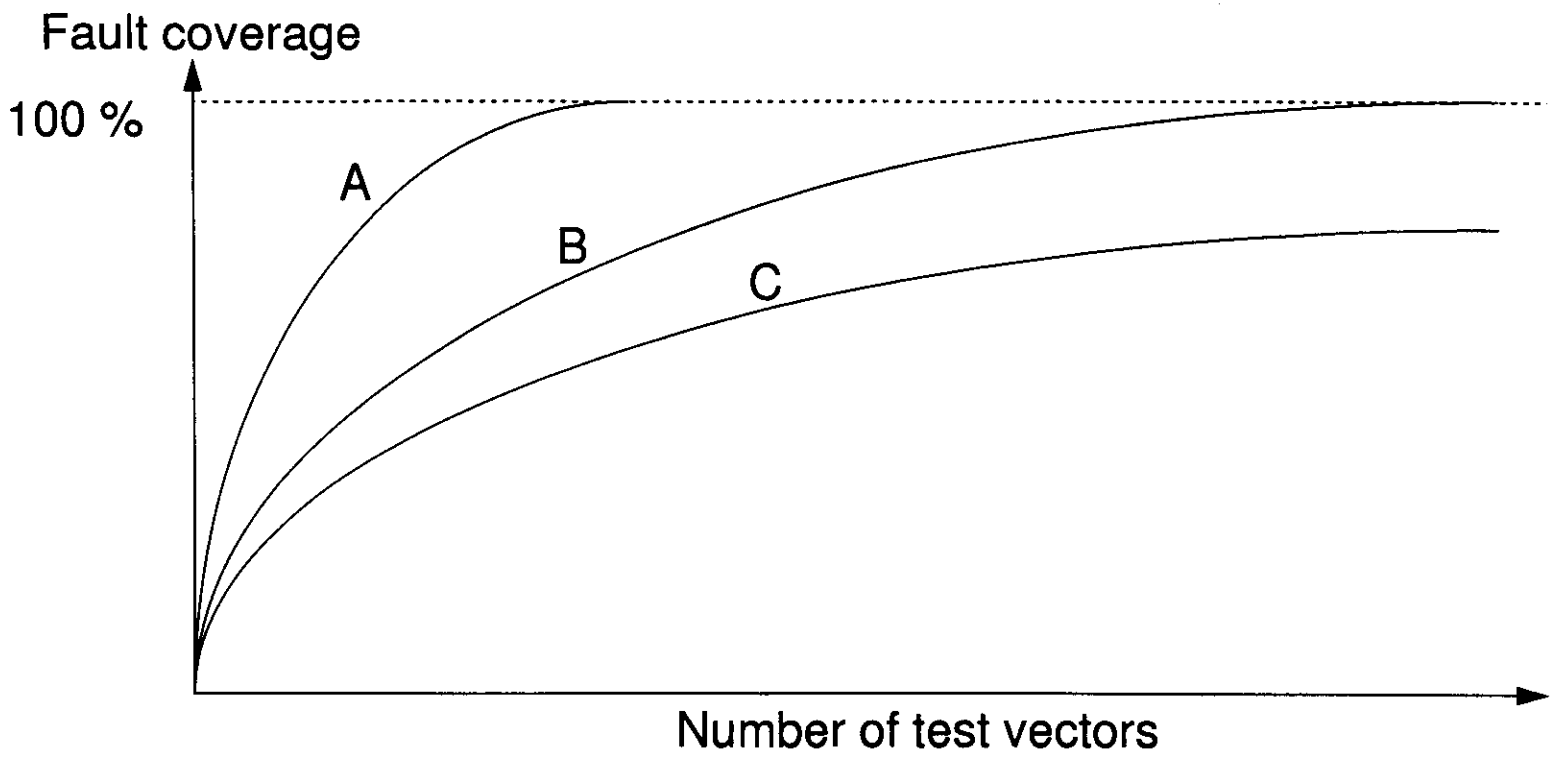
Gate level



- The gate level stuck at 0/1 is the dominantly used fault model for VLSI circuits, because of its simplicity.
- Fault coverage calculated by fault simulation are always calculated using the stuck at 0/1 model. Other more complicated fault models are to compute intensive for VLSI designs.

$$\text{Fault coverage} = \frac{\text{Number of faults detected by test pattern}}{\text{Total number of possible stuck at faults in circuit}}$$

Testability



A: Design made with testability in mind.

B: Design made without testability in mind but good fault coverage obtained by large effort in making test vectors.

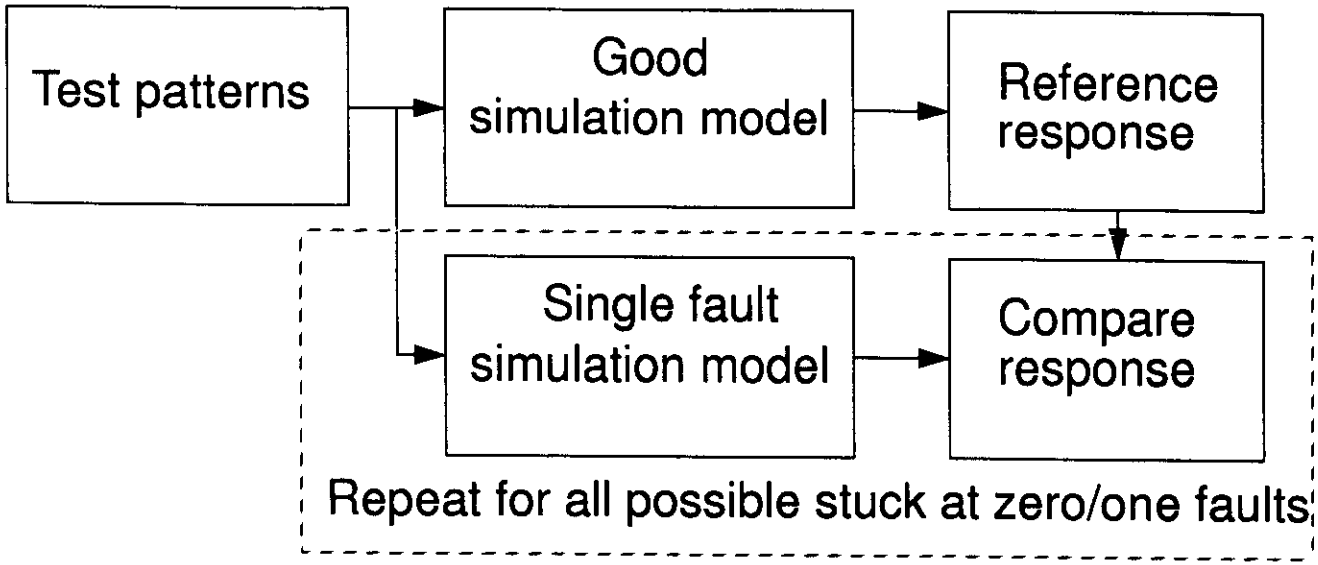
C: Design very difficult to test even using large effort in test vector generation.

Generation of test patterns

- Test vectors made by test engineer based on functional description and schematics. Proprietary test vector languages used to drive tester.
- Testvectors made by design engineer on CAE system.
- Subset of test patterns may be taken from design verification simulations.
- Generated by automatic test pattern generators (ATPG).
- Pseudo random generated test patterns.
- Fault simulation calculates fault coverage.

Fault simulation

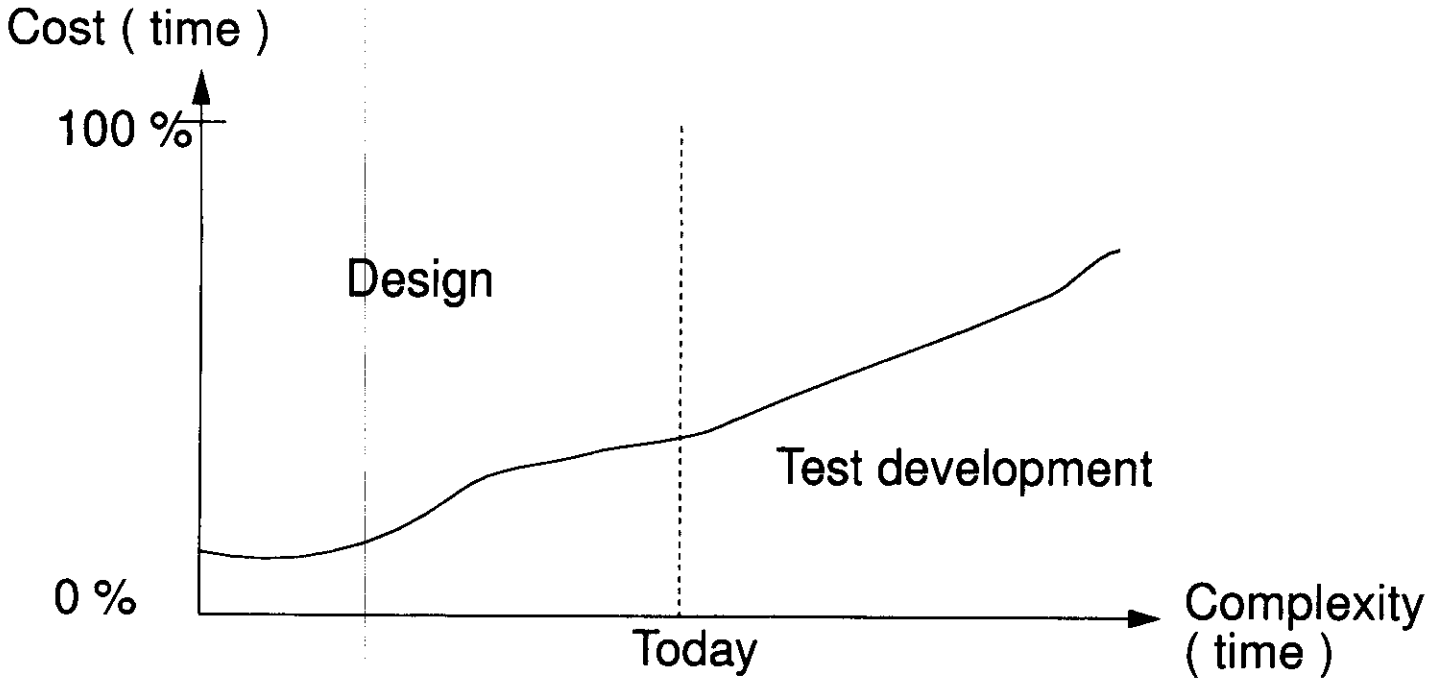
Fault coverage found by fault simulations



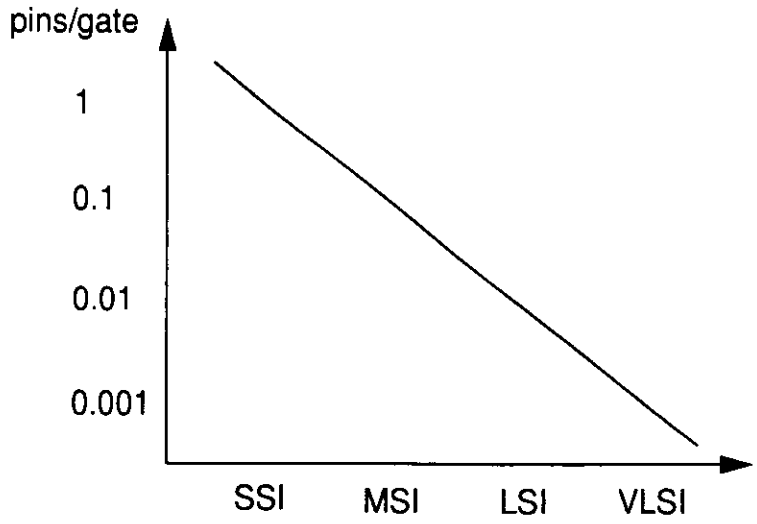
Requires very long simulation times !!.

Toggle test (counts how many times each node has changed) can be used to get a first impression of fault coverage.

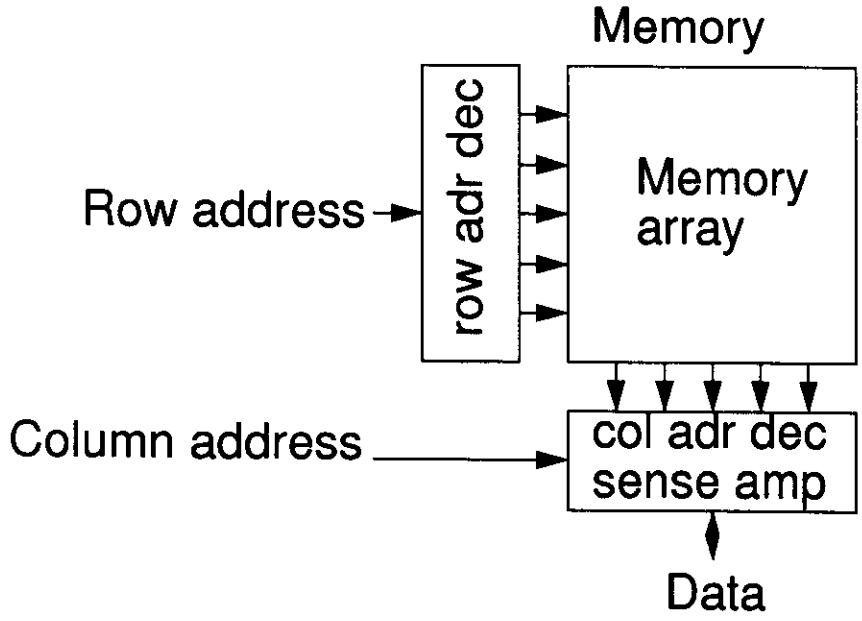
Test development cost when complexity increases:



Testability is decreasing drastically with increased integration level



Memory testing



Exhaustive test of a 1 M memory would take longer than estimated age of our universe.

Algorithmic test patterns used.

Large memory chips have built in redundant memory array columns enabling repair of failing memory cells.

Checker board

0	1	0	1
1	0	1	0
0	1	0	1
1	0	1	0

Test vectors = 4N

Address

0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1

Test vectors = 2N

Walking patterns

1	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

→

0	1	0	0
0	0	0	0
0	0	0	0
0	0	0	0

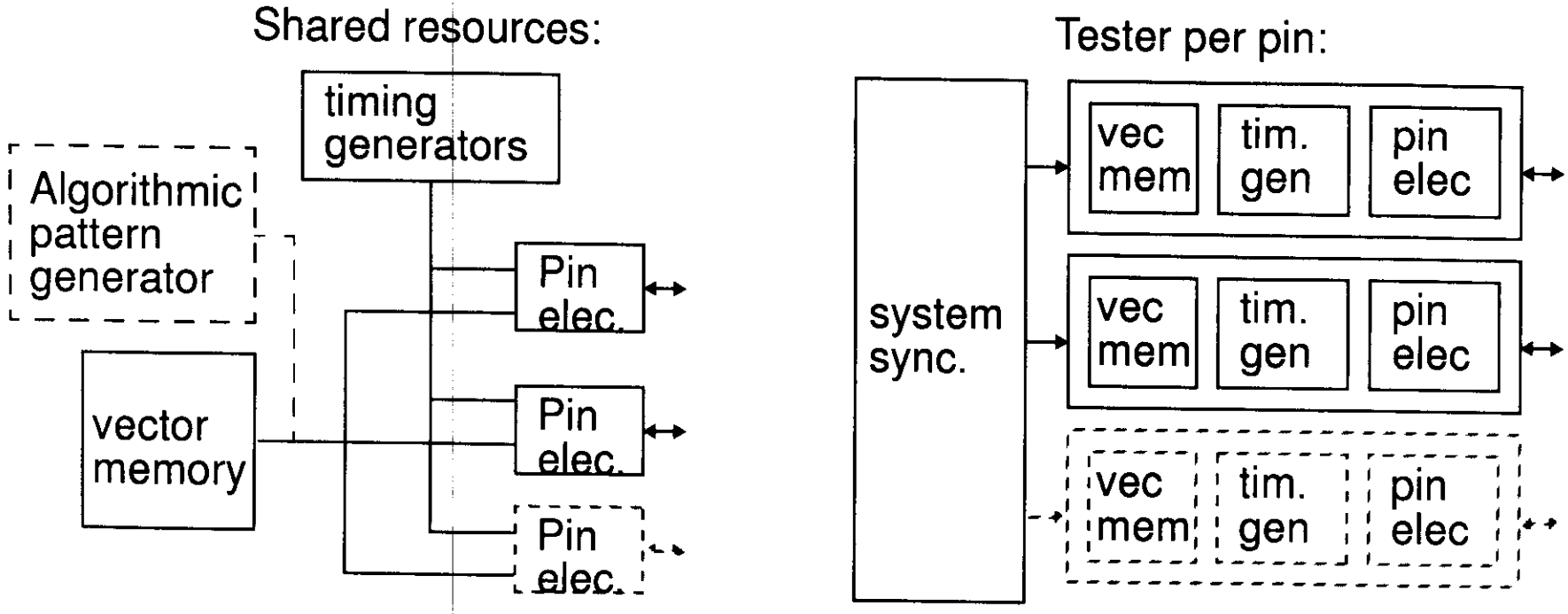
Test vectors = $2N^2$

10 Mhz tester:

64 k = 13 min.
256k = 3.6 hours
1 M = 55 hours

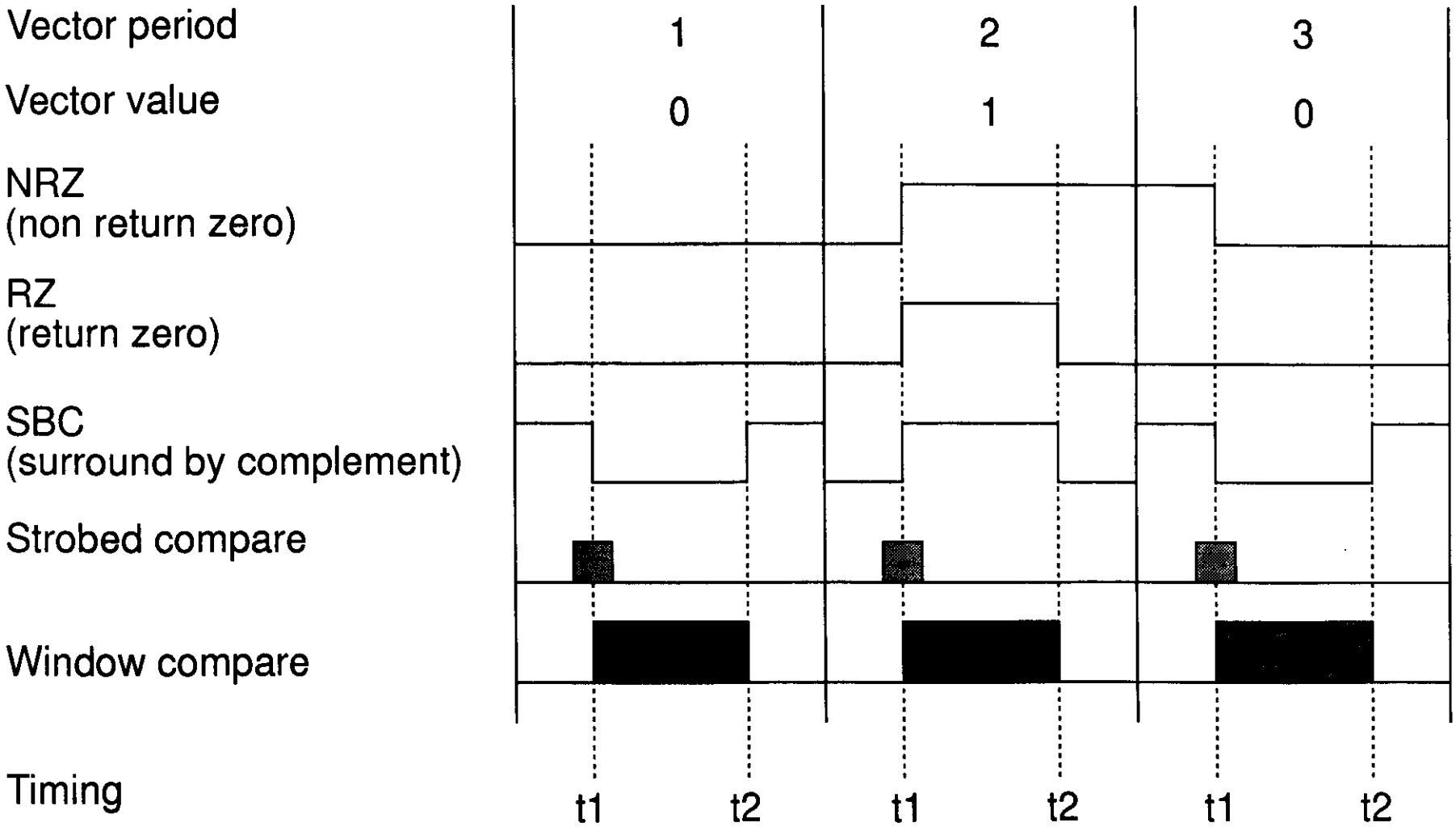
VLSI testers

High speed high pin count VLSI testers are very expensive and complicated machines. (100 k$ - 10 M$).



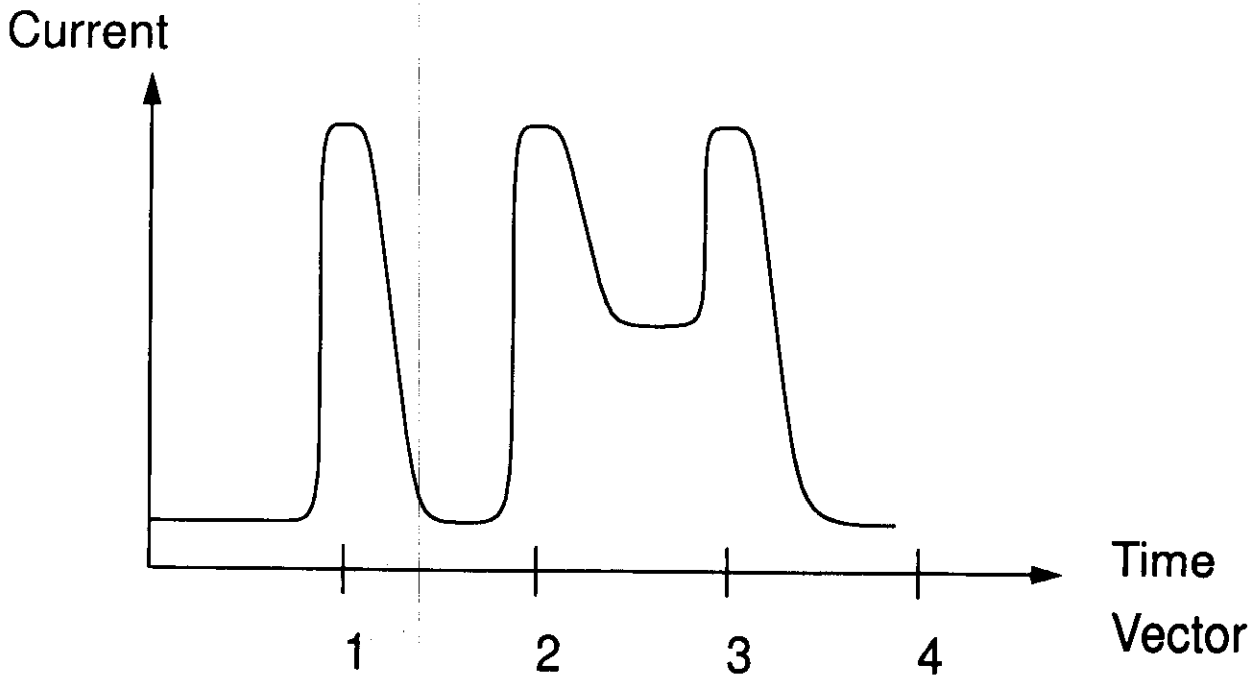
+ Measurements of DC characteristics

Timing formatting of test vectors:



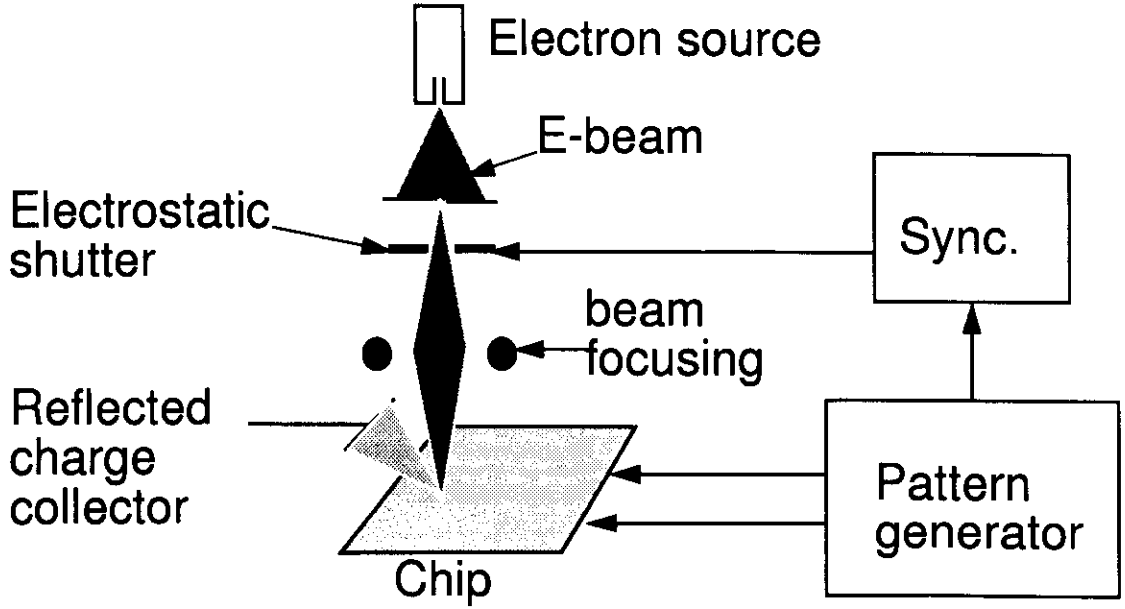
Quiescent current testing

- A CMOS device consumes very low current in steady state.
- If a transistor is stuck on, the steady state current will rise orders of magnitude when the right test pattern is applied



E-beam testing

The reflection of an E-beam from a surface is influenced by voltage potential of the surface.

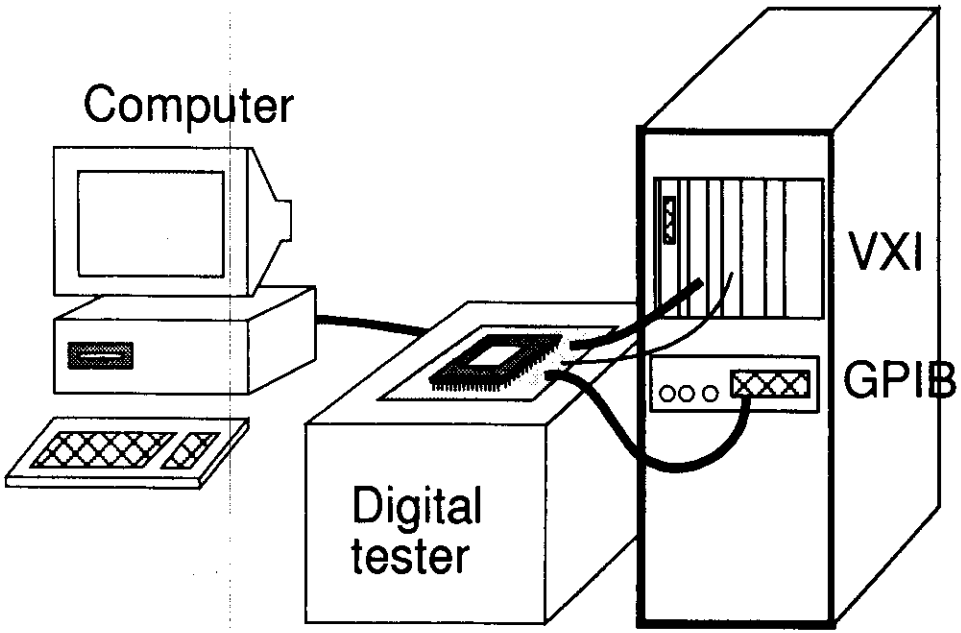


Single point probing with very good timing resolution (~10ps)

Complete scan of chip to get voltage contrast picture at a specific time in pattern sequence.

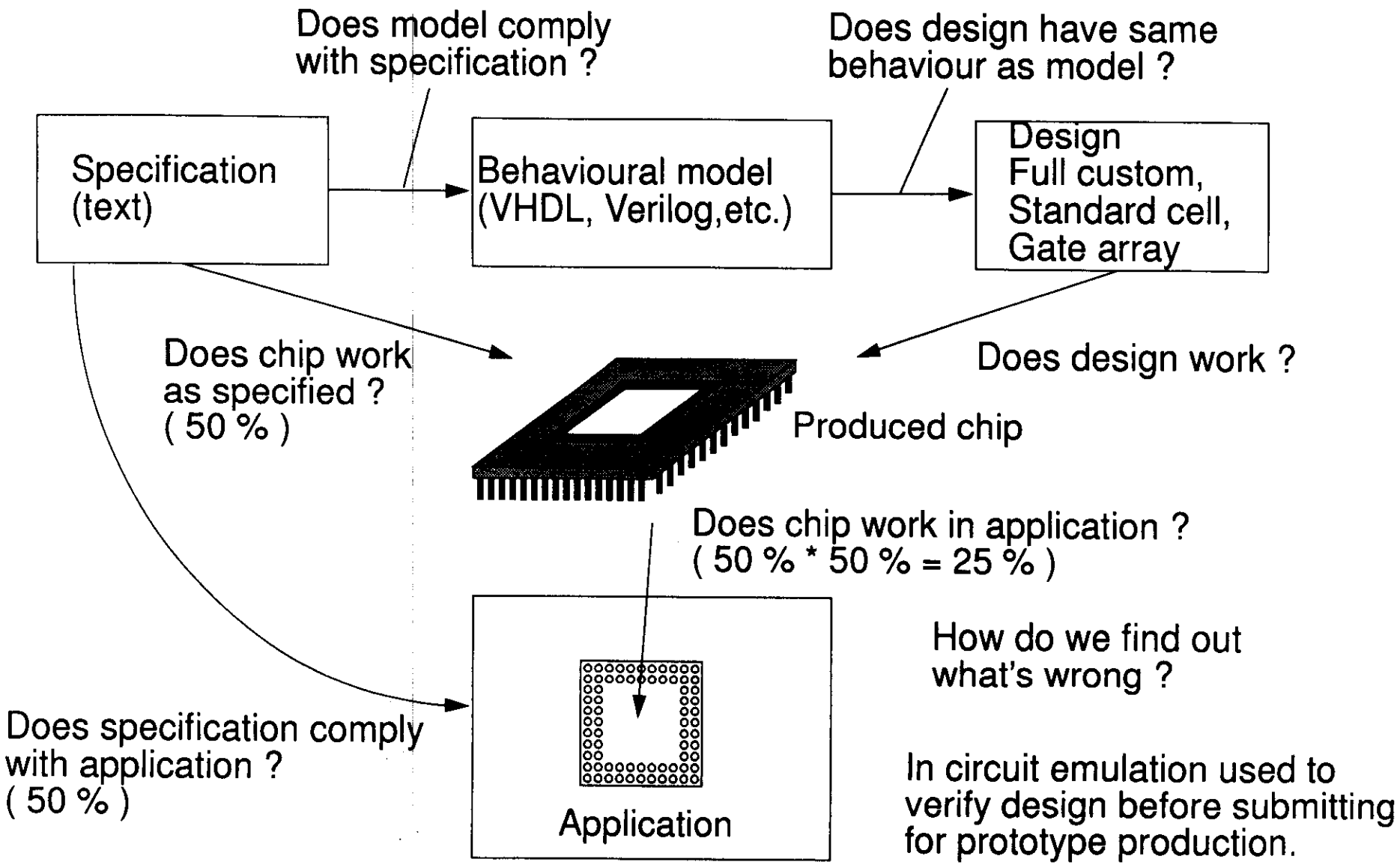
Test of analog circuits

- Each analog circuit is always special.
- Difficult to access internal nodes (drive external load).
- Mixed analog/digital testers are often a digital tester with analog add ons (GPIB, VXI, VME).



Design verification testing

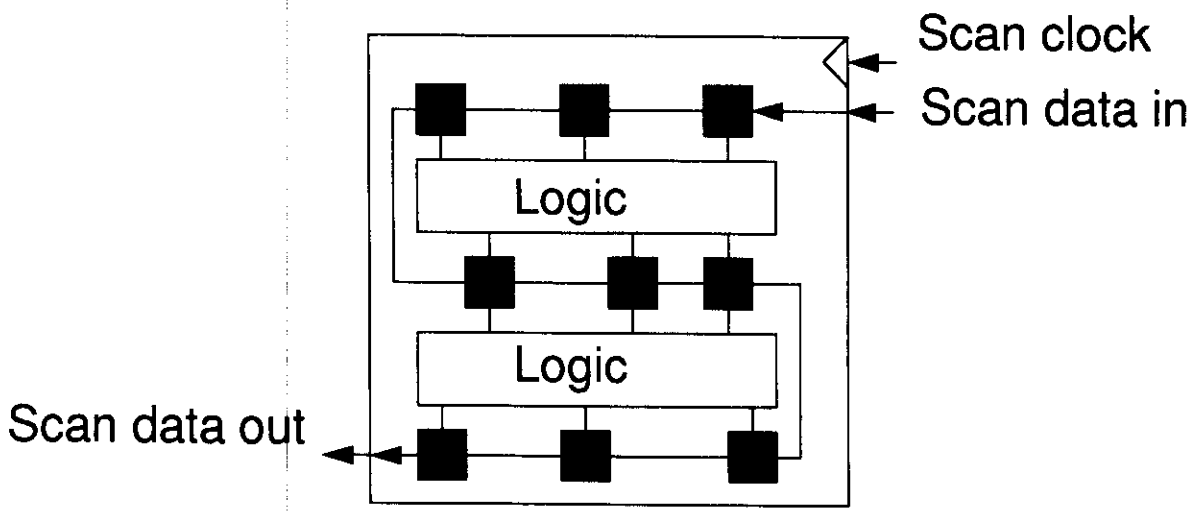
Does our chip really do what it is supposed to do ?



Scan path testing

Scan path testing

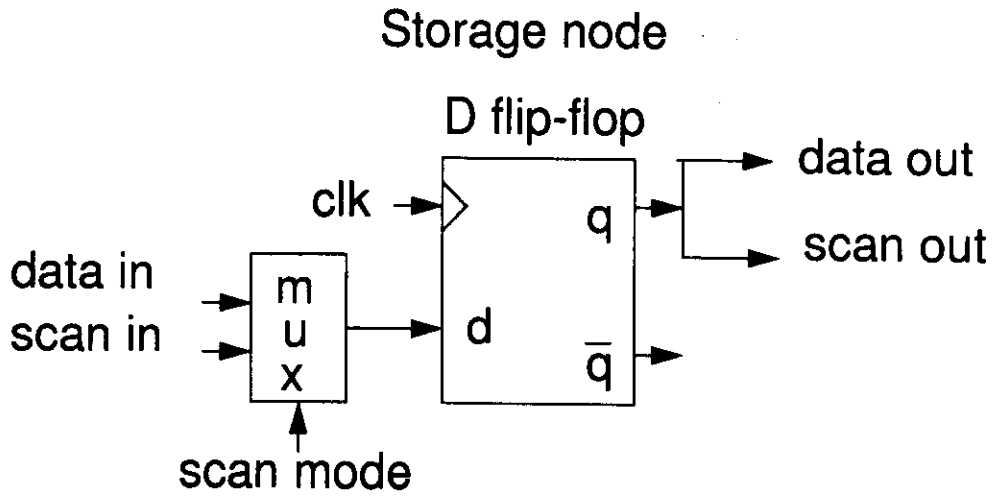
- Improving controllability/observability by enabling all storage nodes to be controlled/observed via serial scan path.

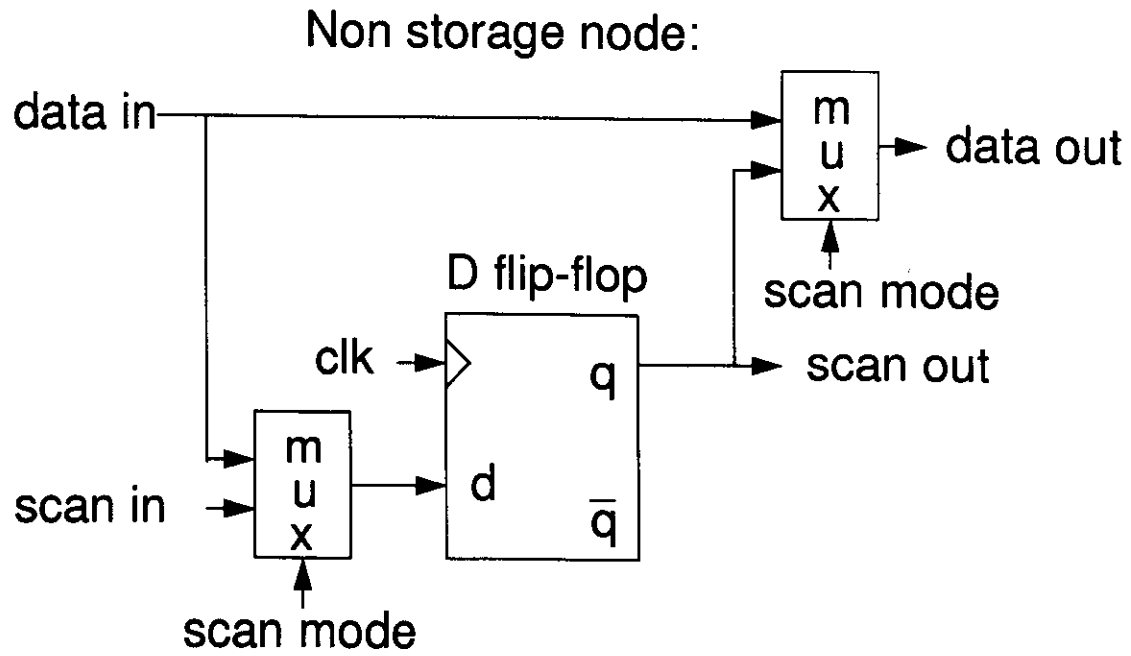


Test principle:
1: Enable scan mode and scan in control data.
2: Disable scan mode and clock chip one cycle.
3: Enable scan mode and scan out observing data.

Generation of test vectors: With the high controllability/observability the test vectors can be generated automatically with a ATPG program.

Scan path cells:





- Scan path advantages:

  Test vectors can be generated by ATPG programs.

  Observability/Controllability problems do not have to be considered during the design phase.

  Testers do not need to have complex test vector generation capabilities for all pins of the chip (only scan in and scan out necessary).

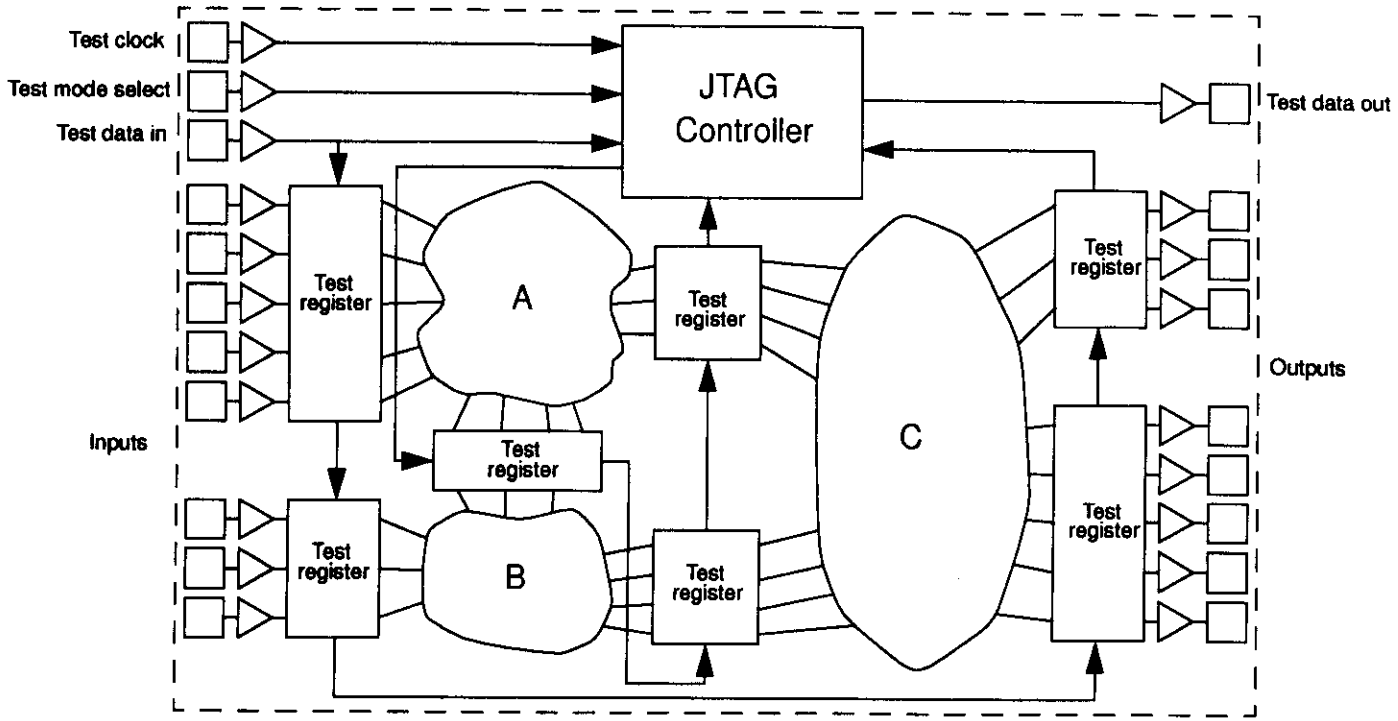
- Scan path disadvantages:

  Hardware overhead: additional multiplexers must be included in the circuit.
  example: 20.000 gates with 500 flip-flops
  1 flip-flop = 10 gates > 500 ff = 5000 gates
  1 scan flip-flop = 14 gates > 500 ff = 7000 gates.
  overhead = 2000 gates = 10%

  Speed degrading: additional multiplexers added in signal path.
  example: 2 input inverting multiplexer in 1 µm CMOS dly= 0.44 ns (typ.).
  special scan flip-flop in 1µm CMOS dly = 0.3 ns (typ.).

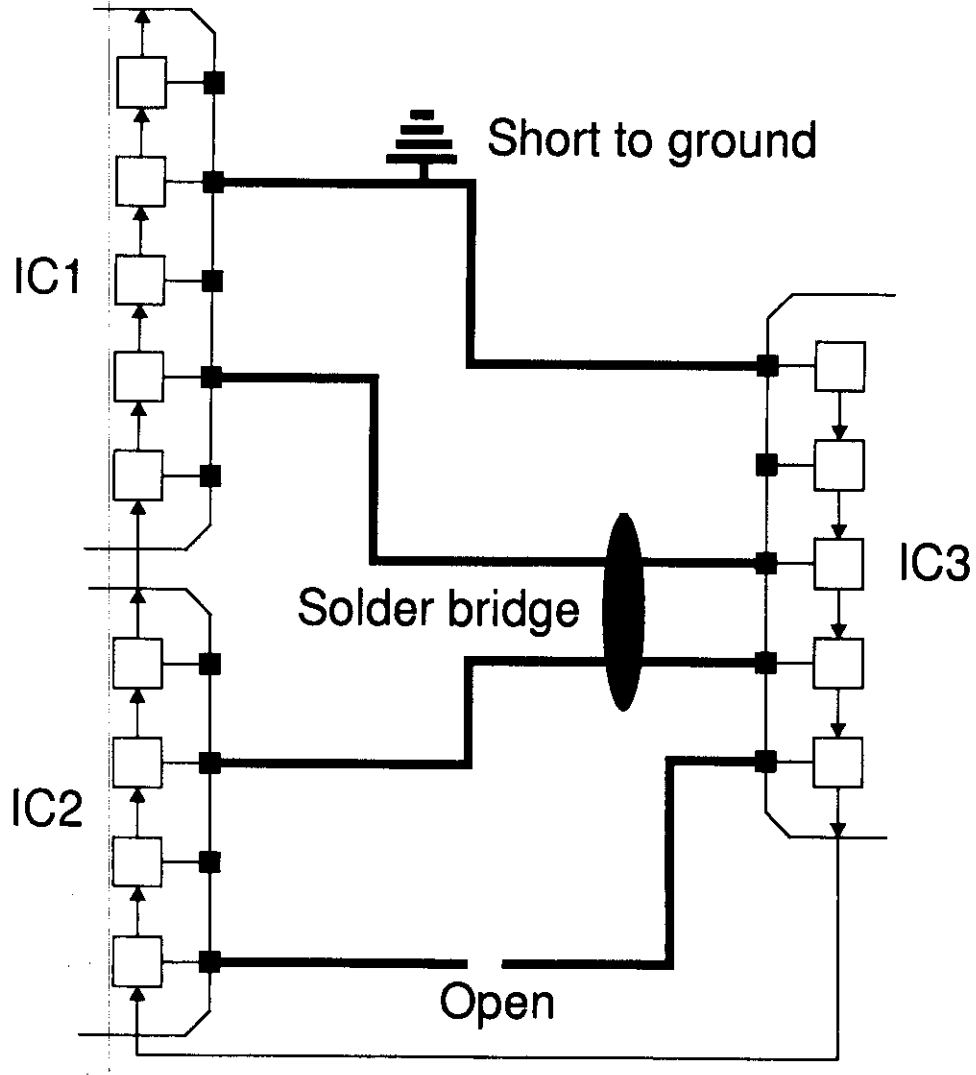
The JTAG standard

- IEEE 1149 standard.
- Boundary scan to test interconnect between chips.
- Internal scan to test chip.
- Control and status of built in self test.
- Chip ID
- Several commercial chips with JTAG standard implemented: Motorola 68040, Texas DSP processors. etc.



Boundary scan makes it possible to test interconnections between chips on a module.

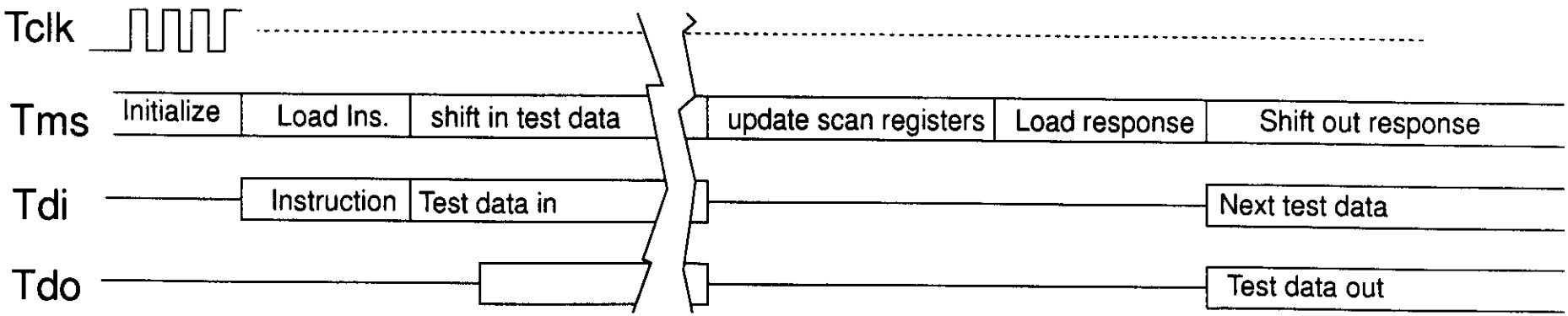
Test of chips and board connections can be performed in-situ.



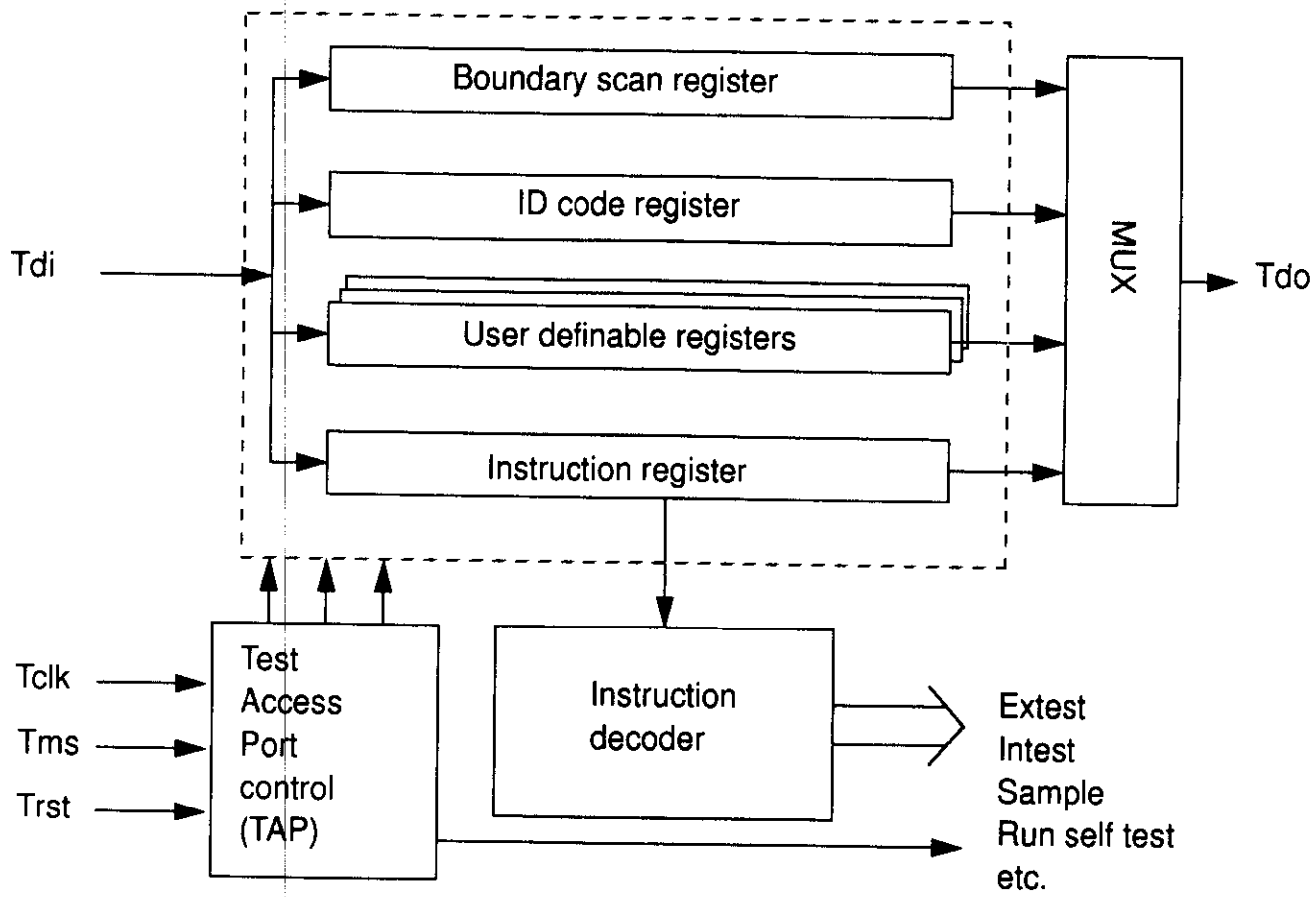
JTAG Protocol

Only 4 (5) pins used for JTAG interface

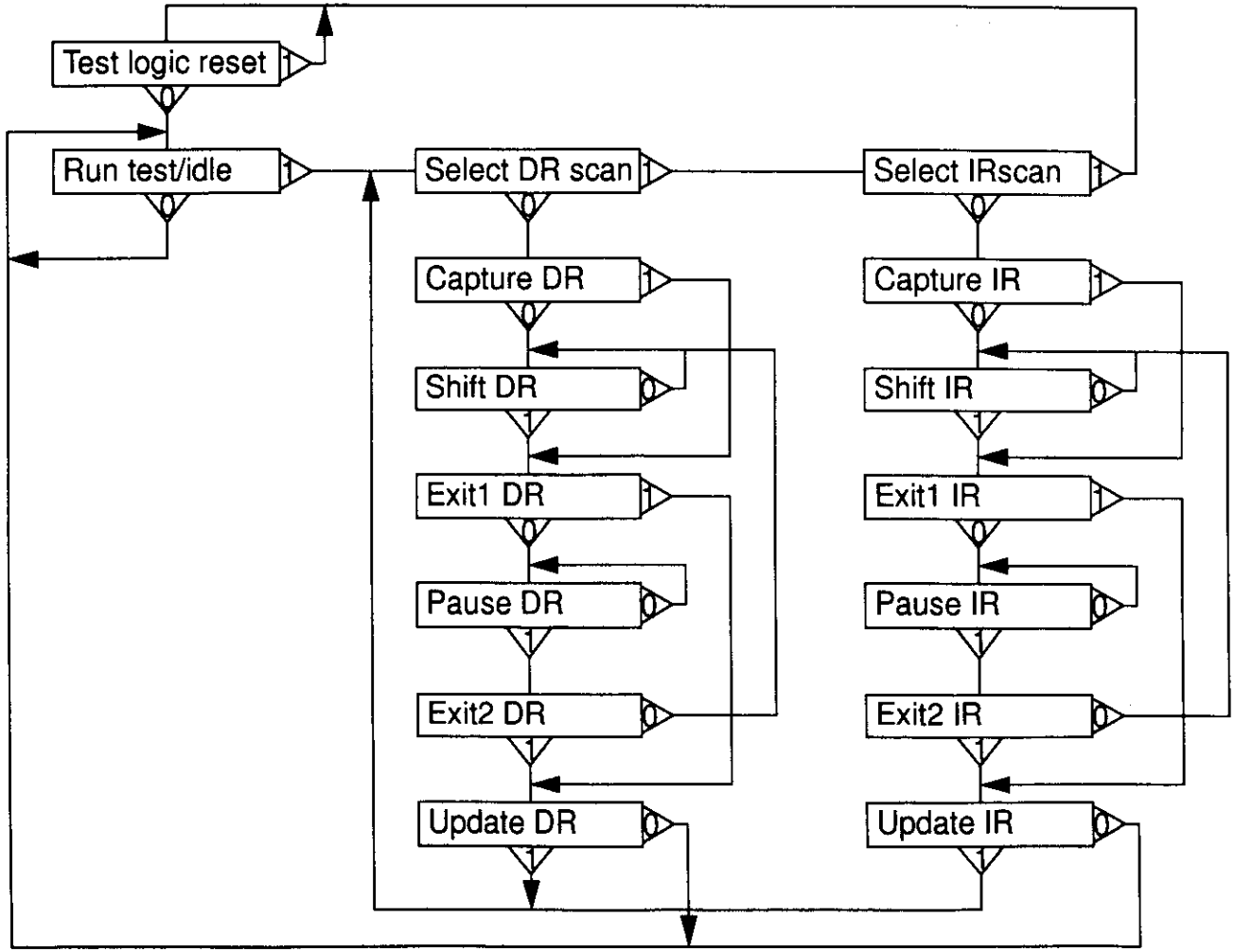
Test clock: Clock for loading control and test patterns + clock for shifting out response
Test mode select: Selects mode of testing
Test data in: Serial input of test patterns
Test data out: serial output of response to test pattern.



JTAG block diagram



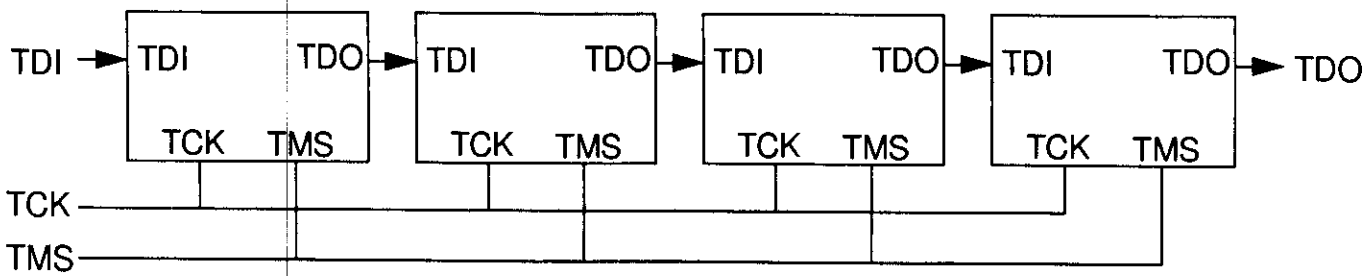
JTAG Test Access Port (TAP) controller



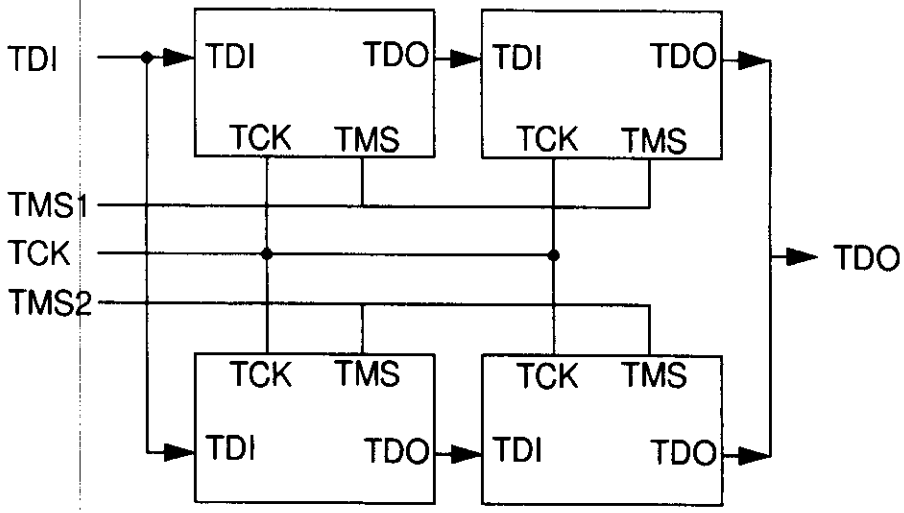
TAP state transition only depends on Tms

If Tms keept at logic one TAP controller will get to Test-logic-reset state

Connection of IC's with JTAG

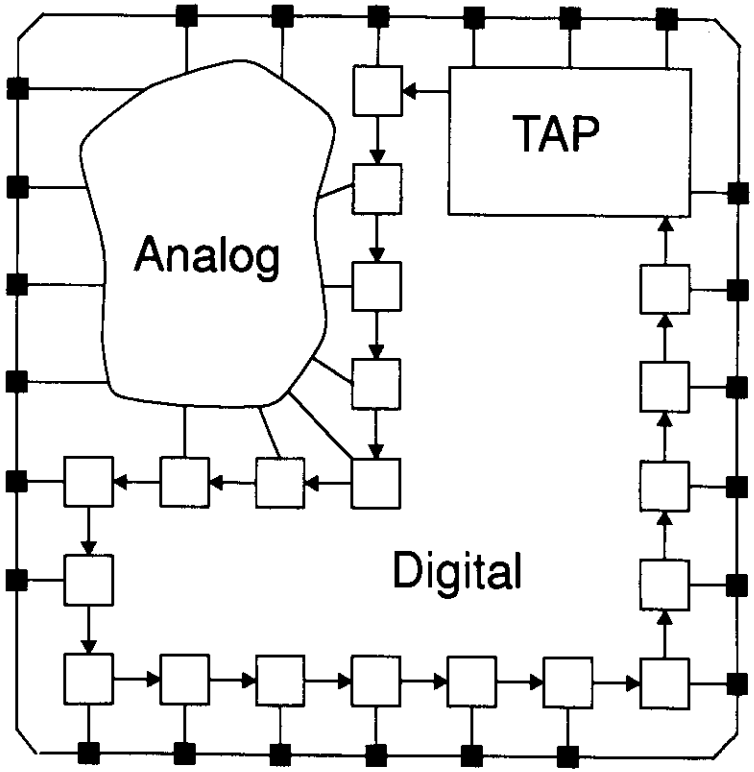


Serial connection



Hybrid serial/parallel connection

Using JTAG testing of mixed analog/digital IC's

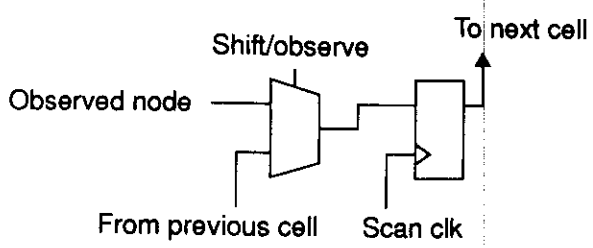


Consider analog part as being external and insert boundary scan registers between analog and digital.

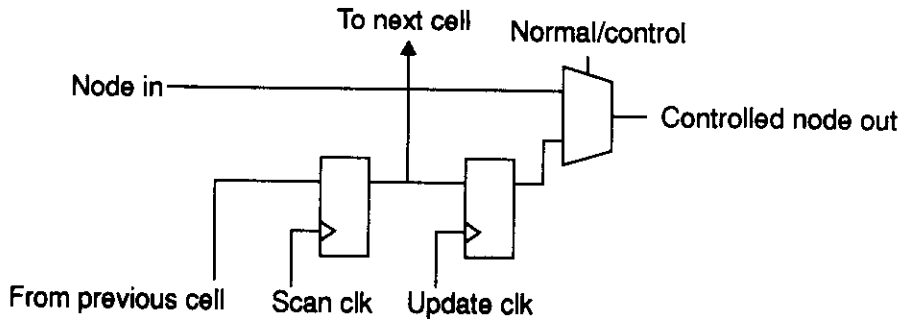
New IEEE 1149.4 standard for test of analog parts in the process of being defined.

JTAG scan cells

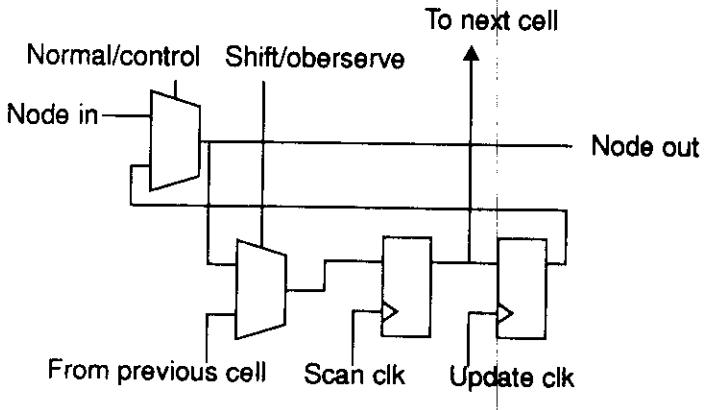
Observing scan cell



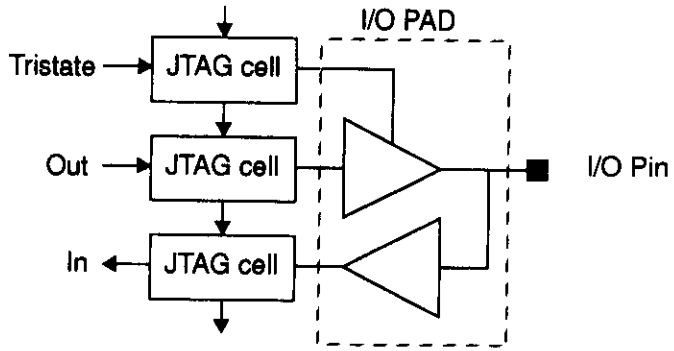
Controlling scan cell



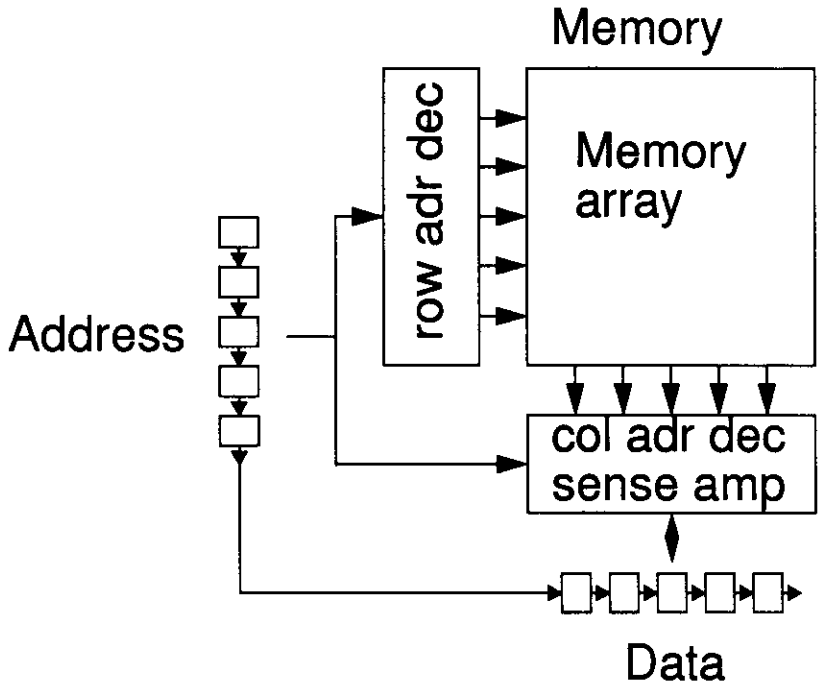
Observing and controlling scan cell



JTAG cells required for Input/Output pin



JTAG testing of embedded on-chip memories



Each memory test vector must be shifted in/out serially

⇓

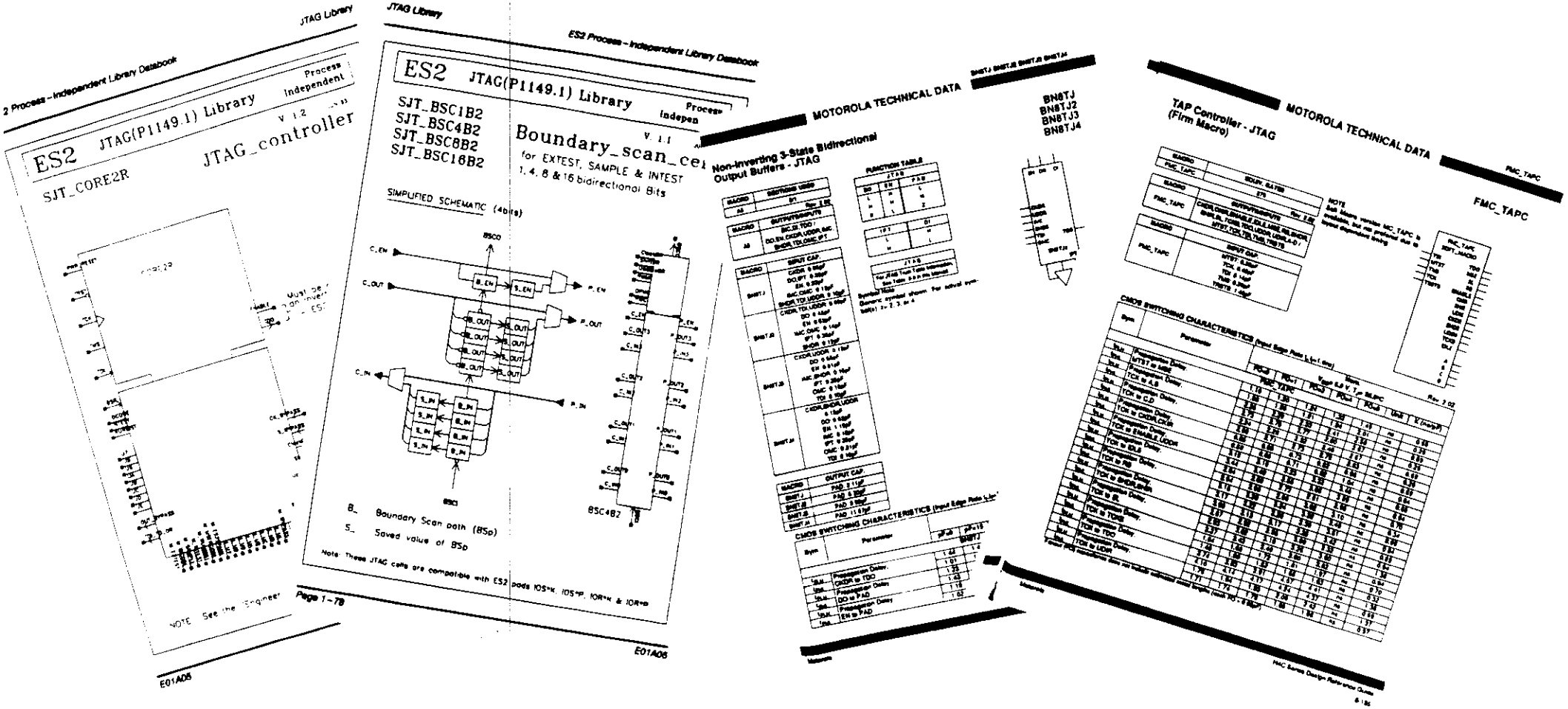
Testing becomes very, very, very SLOW

⇓

Use special built in self test

JTAG libraries from ASIC vendors

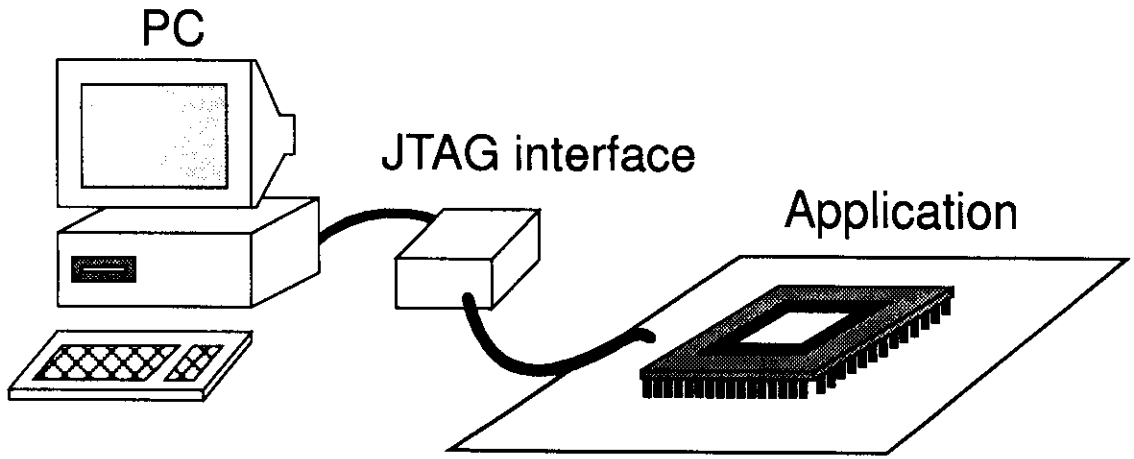
Libraries of JTAG components are normally availeble when designing with standard cells or gate arrays.



JTAG test equipment

Most chip testers today have options of special JTAG test facilities.

Cheap PC based JTAG test equipment today available at attractive prices.



Software:

Test vector interface
Netlist interface (EDIF)
Scan path description interface (Boundary scan description language)
JTAG test functions
Fault diagnostics
(Automatic test pattern generation for inter chip connections)
Etc.

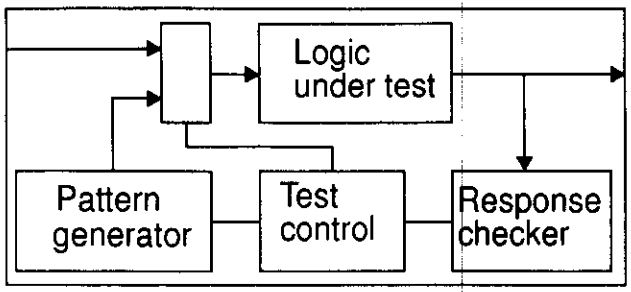
Alternative use of JTAG

- Load programming data into programmable devices before use.
- Monitor function of device while running.
- Read out of internal registers in micro processors and digital signal processors to ease debugging of programs.

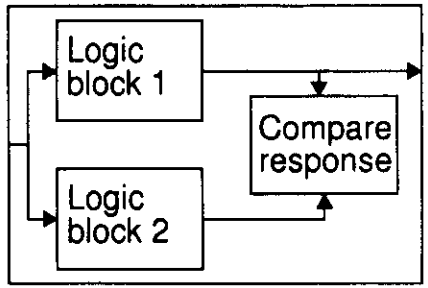
Built in test

Different schemes of built in (self) test

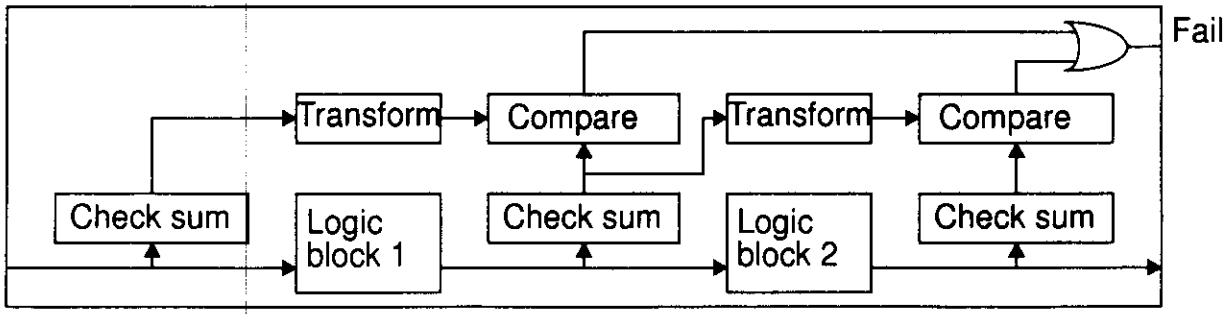
Include test pattern generator and response check on chip



Make self checking during operation by duplicating all functions



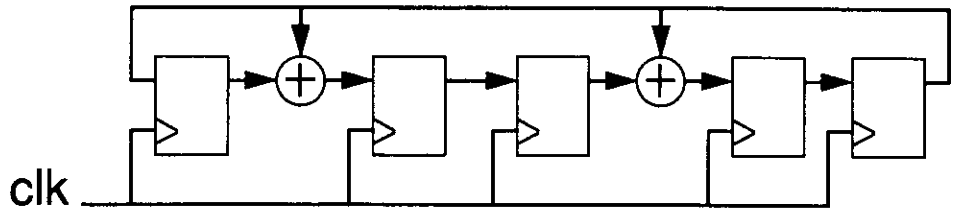
Generate local check sums and check with transformation of previous check sum



Hardware overhead !!

Simple pattern generation and pattern checking

Linear Feedback shift register (LFSR)

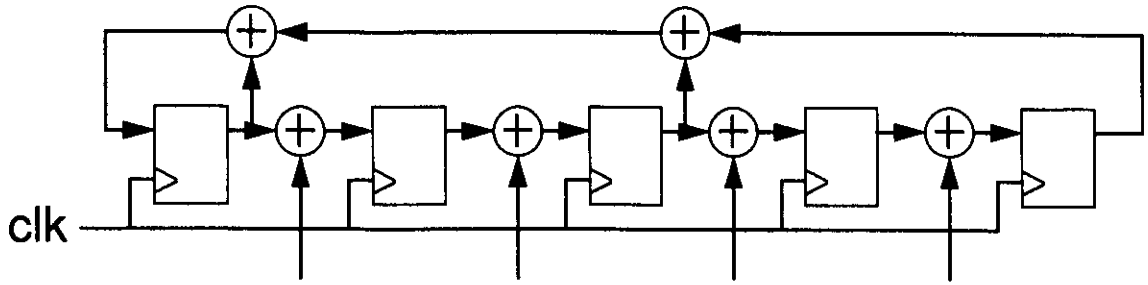


Based on polynomial division.

⊕ = exclusive or

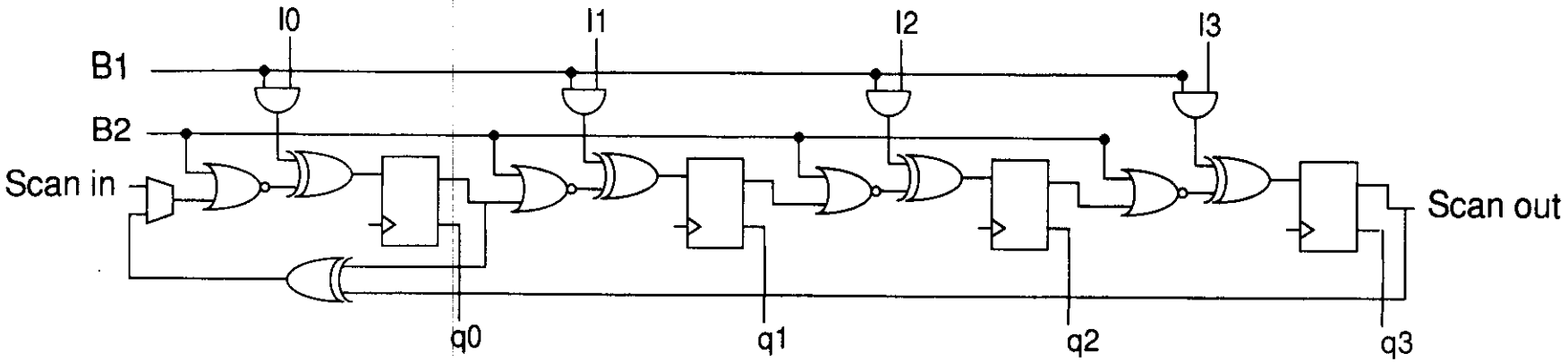
Pattern generation: Pseudo random patterns based on generating polynomial and seed.

Pattern checking: Multiple input signature register (MISR) generating "check sum" of input data.

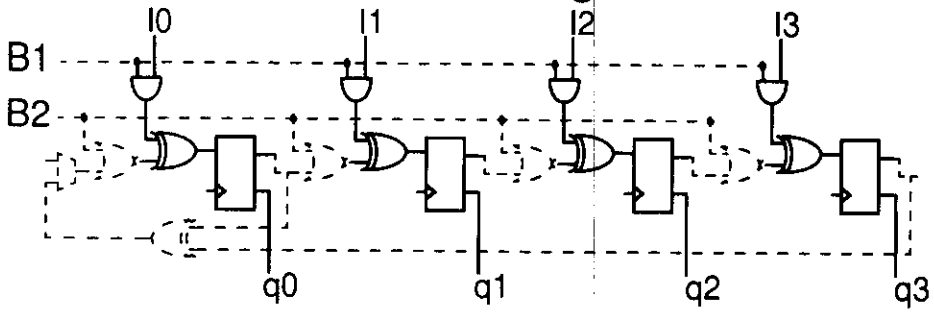


Scan path cells can be implemented so they can be used as pseudo random pattern generator or multiple input signature analysing register.

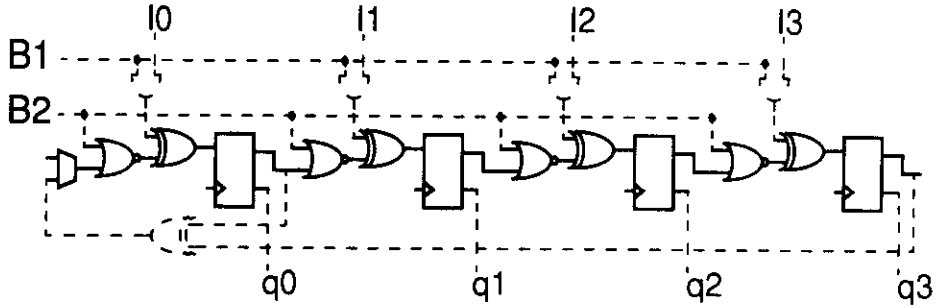
BILBO (Built In Logic Block Observer)



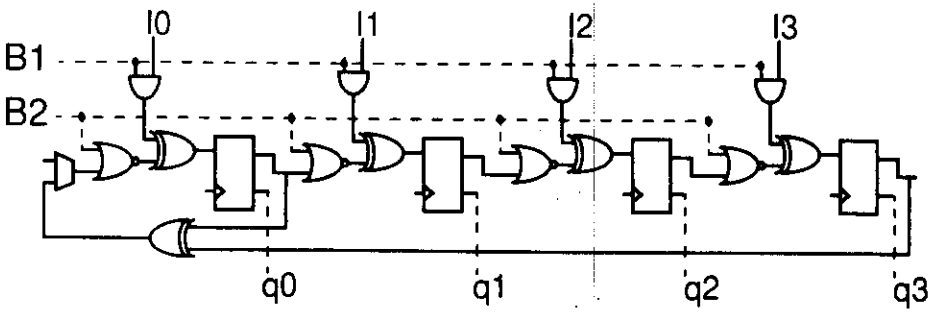
B1,B2 = 11, Normal register mode



B1,B2 = 00, Scan path mode



B1,B2 = 10, LFSR mode



B1,B2 = 01, Reset of BILBO

Design for testability guidelines.

- Use static logic.
- Make design completely synchronous.
  use D flip-flops and not latches.
  no clock gating.
- No internal clock generation.
- Prevent large counter like structures.
- If possible use scan path (JTAG).
- If possible use built in test of memories.

DO NOT FORGET ABOUT TESTING WHEN CHIP IS SPECIFIED AND DESIGNED

Testing seen from an ASIC designer.

- Design verification simulations performed at full speed.
- Functional testing performed at low speed (1 Mhz).
- Few timing path delays performed to monitor process.
- Single quiescent current measurement.
- Test structures on wafers used to monitor process.
- Test vectors taken from design verification simulations.
- Test vectors must conform to tester restrictions.
  (checked by special programs)
- Most ASIC manufactures offer scan path cells and ATPG programs.
- Most ASIC manufactures offer JTAG boundary scan I/O cells and TAP controller.

Cheap and easy to use chip design tool set

Functions in top of the line CAE tools

- Hierarchical Schematic entry.
- Hierarchical layout editor.
- Hardware describing language + simulator (VHDL).
- Layout extractor.
- Design rule check.
- Electrical rule check.
- Layout versus schematic check.
- Netlist generator.
- Analog simulator (Spice).
- Place and route (standard cell, gate arrays).
- Test vector generation + tester interface.

Cheap and easy to use chip design tool set

- Design transfer interface (GDSII, CIF)
- Logic synthesis and optimization.
- Transistor level optimization.
- Micro wave extraction and simulation.
- Layout synthesis from transistor schematics.
- Parameterized cell generator.
- Array generator (memories, adders, multipliers).
- Data path generators.
- Plotter interfaces.
- Automatic test pattern generator.
- Fault simulation.
- Netlist and schematic interface (EDIF)

Implications of using top of the line tools

- A complete set of chip design tools costs up to 500.000 $ per seat (list price).
- Power full workstation with 100 MB memory and several Giga bytes of hard disk required.
- Complicated tools may have severe bugs which may be difficult to find a way around.
- New releases of tools solves old bugs but normally also introduce new unknown bugs.
- Large patches may be required to get specific functions to work as required.
- Configuration management and setup of tools requires substantial effort.
- 1/2 - 1 full time person required to keep set of tools working and up to date.

Required technology information

- Design manual.
- Design kit.
- Process layer definitions.
- Design rules for DRC.
- Extraction rules for extractor.
- Simulation models (Spice, VHDL, Verilog).
- Back annotation timing parameters.
- Standard cell library.
- Place and route setup.
- Test vector interface.

Cheap and easy to use PC tool (Ledit from Tanner)

- Have all basic functions required:
  Schematic entry, Layout editor, Extraction, Design rule check, Layout versus schematic, Spice simulator, Place & route, Memory generator.
- Requires only PC.
- Runs under DOS.
- Several technologies delivered with system.
- Cheap (Universities: aprox. 1000 $ for basic kit).

Public domain software (MAGIC).