Chapter 1 Introduction to VLSI Testing

超大型積體電路測試簡介

趙家佐

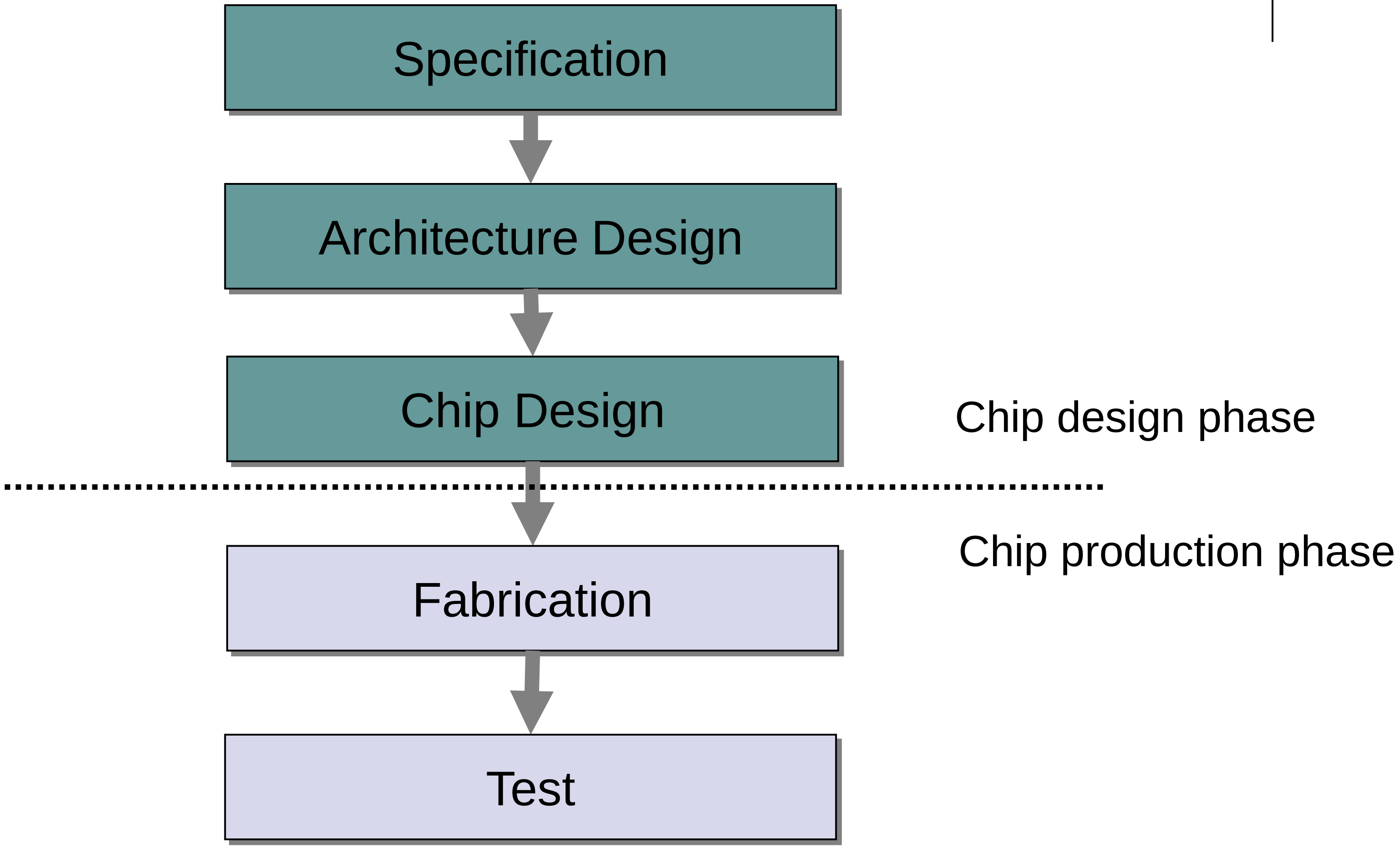
Goal of this Lecture

- Understand the process of testing
- Familiar with terms used in testing
- View testing as a problem of economics

Introduction to IC Testing

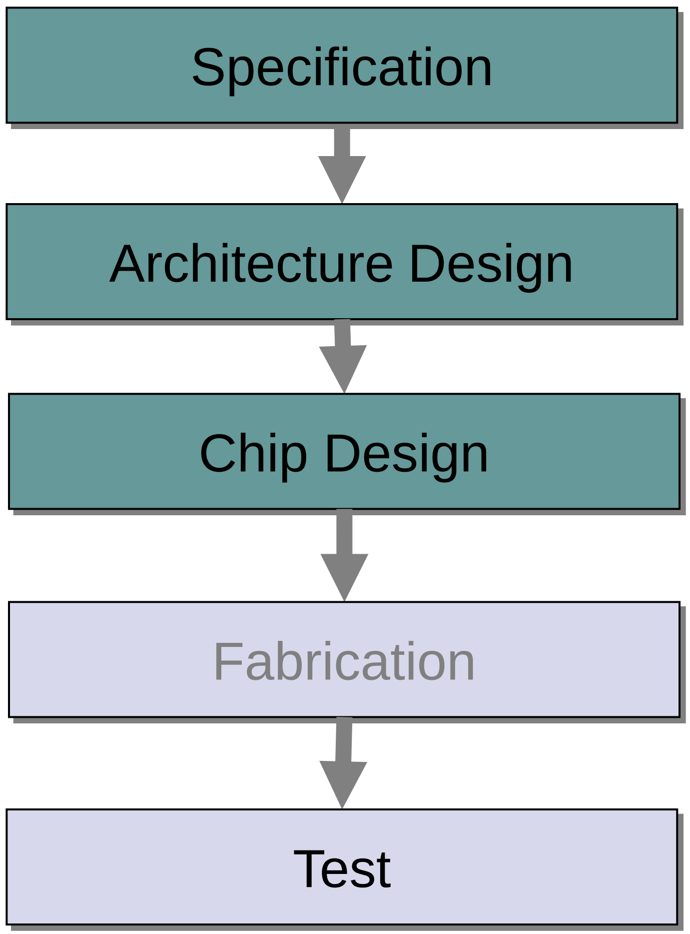
- Introduction of design flow and testing
- Types of IC testing
- Manufacturing tests
- Test quality and economy
- Test industry

IC (SOC) Design/manufacture Process



- In chip production, every manufactured chip will be tested.
- A chip is shipped to customers, if it works according to specification.

Tasks of IC Design Phase



- **Function and performance requirements**
- **Die size estimation**
- **Power analysis**
- **Early IO assignment**

- **High-Level Description Block diagrams**
- **IP/Cores selection (mapped to a platform)**
- **SW/HW partition/designs**

- **Logic synthesis**
- **Timing verification**
- **Placement, route and layout**
- **Physical synthesis**
- **Test development and plan**

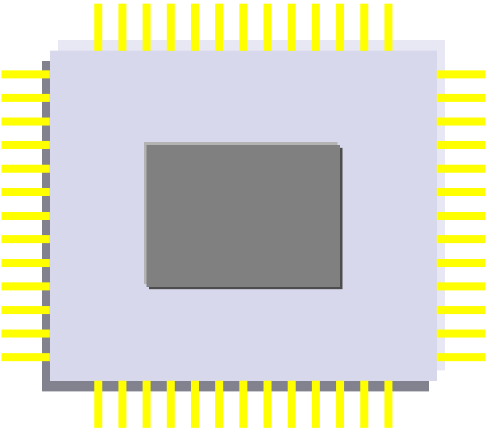
- **First silicon debug**
- **Characterization**
- **Production tests**

Objectives of VLSI Testing

- Exercise the system and analyze the response to ascertain whether it behaves correctly after manufacturing
- Test objectives
  - Ensure product quality
  - Diagnosis & repair
- All considered under the constraints of economics

Test Challenges

- Test time exploded for exhaustive testing
  - For a combinational circuit with 50 inputs, we need $2^{50} = 1.126\mathbf{x}10^{15}$ patterns = $1.125\mathbf{x}10^{8}$s = 3.57yrs. (10^{-7}s/pattern)
  - Combinational circuit = circuit without memory



Too many input pins → too many input patterns

More Challenges

- High automatic test equipment (ATE) cost for functional tests
- Testing circuits with high clock rates
- Effect of advanced process
  - Crosstalk, power, leakage, lithography, high vth variation…
- Test power > design power
- Integration of analog/digital/memories
- SOC complexities

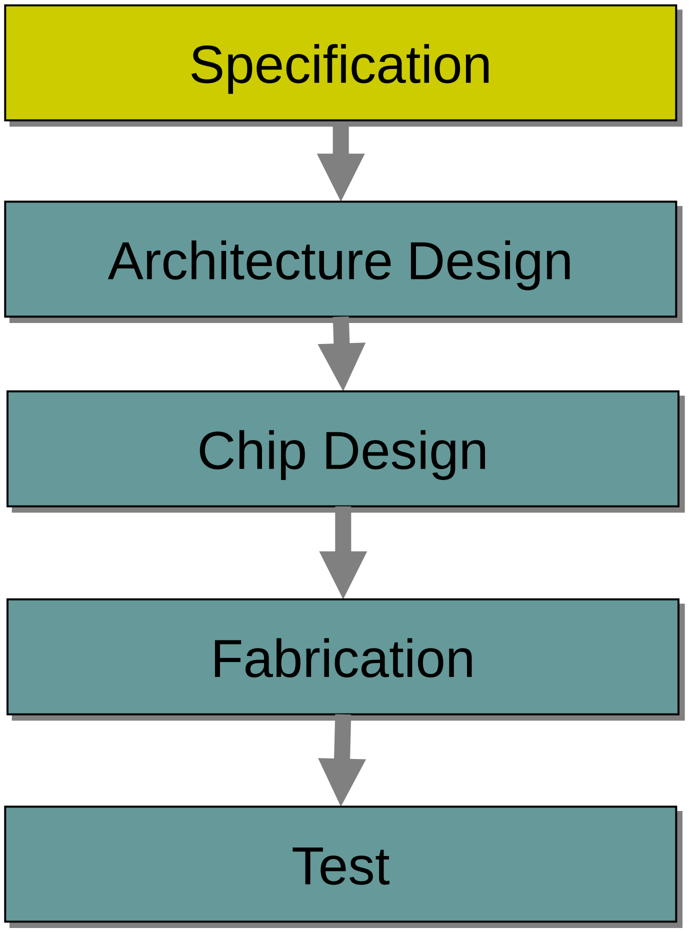
Testing Cost

- Test equipment cost
  - Analog/digital signal and measuring instrumentation
  - Test head
  - Test controller (computer & storage)
- Test development cost
  - Test planning, test program development and debug
- Testing-time cost
  - Time using the equipment to support testing
- Test personnel cost
  - Training/working

Testing Cost

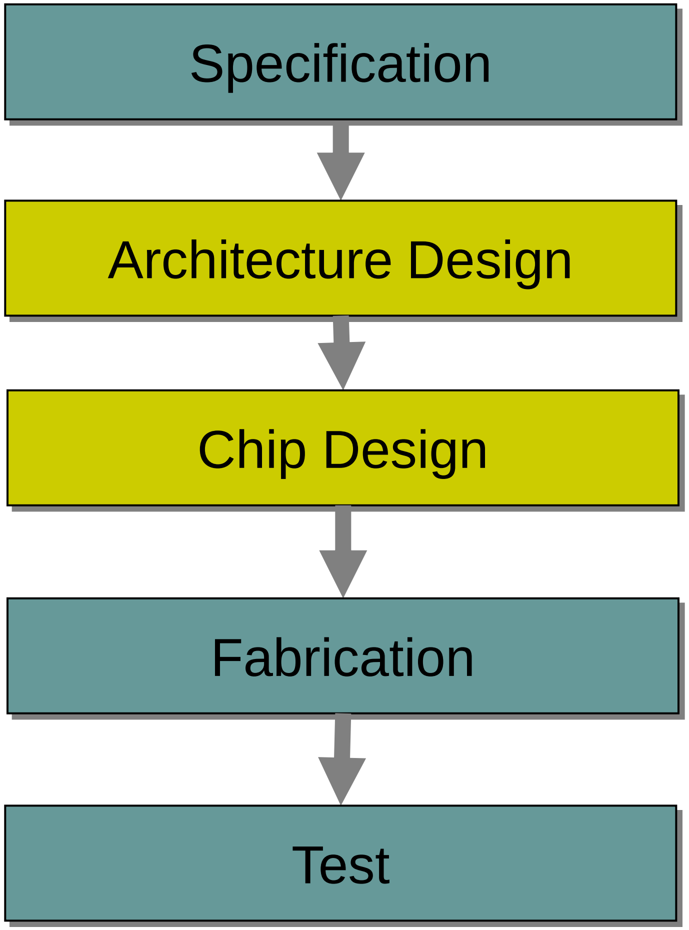
- Testing of complex IC is responsible for the second highest contribution to the total manufacturing cost (after wafer fabrication)
- 0.5-1.0GHz, analog instruments, 1024 digital pins: ATE purchase price
  - $1.2M + 1024*$3000 = $4.272M
- Running cost (5-yr linear depreciation)
  - = Depreciation + Maintenance + Operation
  - = $0.854M + $0.085M + $0.5M
  - = $1.439M/yr

Types of IC Testing (I): Audition of System Specification



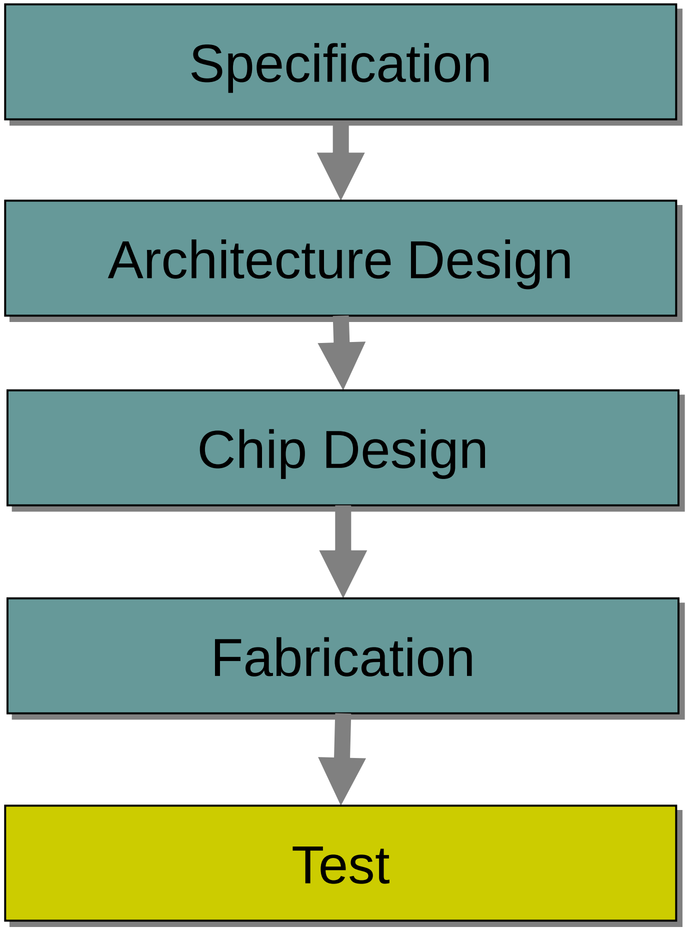
- Translation of customer requirements to system specifications is audited.
- The specification has to be reviewed carefully throughout the design/production process.

Types of IC Testing (II): Verification



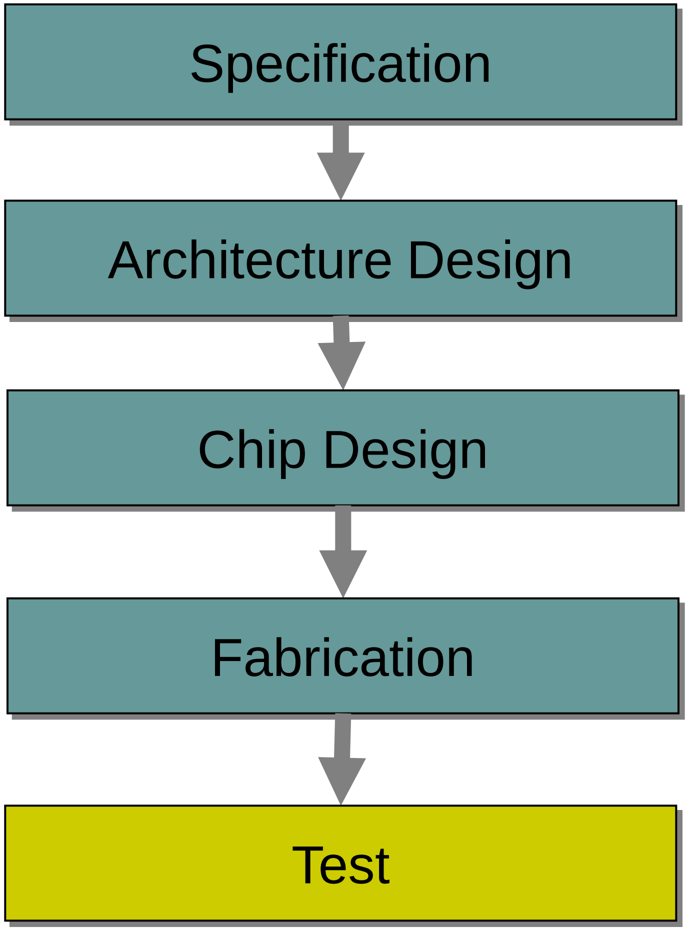
- The design is verified against the system specifications to ensure its correctness.
- Verification is an essential and integral part of the design process.
- Especially for complex designs, the time and resource for verification exceed those allocated for design.

Types of IC Testing (III): Characterization Testing



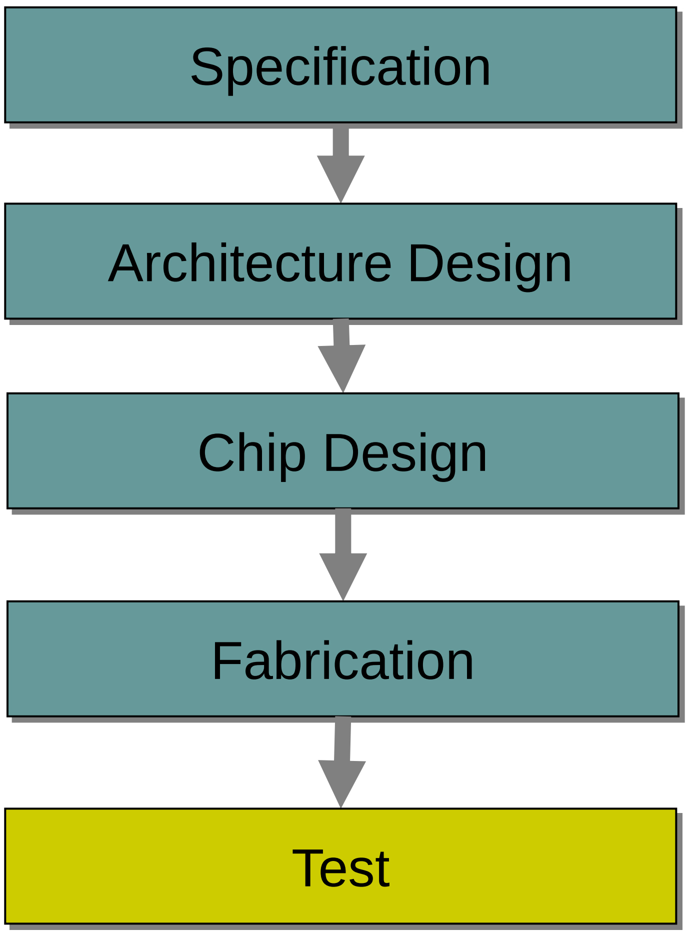
- Before production, characterization testing are used.
  - Design debug and verification.
  - Determine the characteristics of chips in silicon.
  - Setup final specifications and production tests.

Types of IC Testing (IV): Production Testing



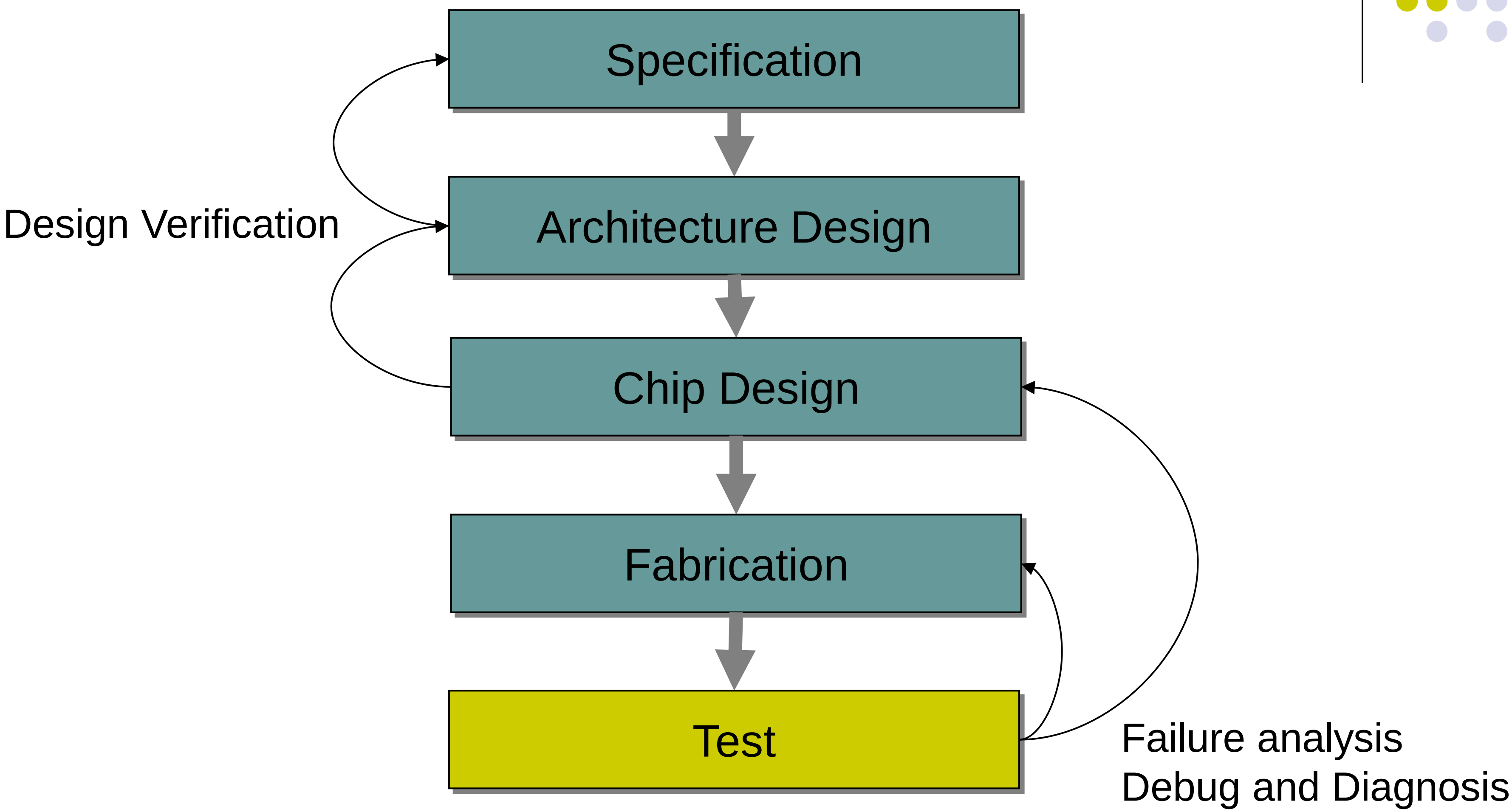
- In production, all fabricated parts are subjected to production testing to detect process defects.
- To enforce quality requirements
  - Applied to every fabricated part
  - The test set is short but verifies all relevant specifications, i.e., high coverage of modeled faults
- Test cost and time are the main drivers.

Types of IC Testing (V): Diagnosis



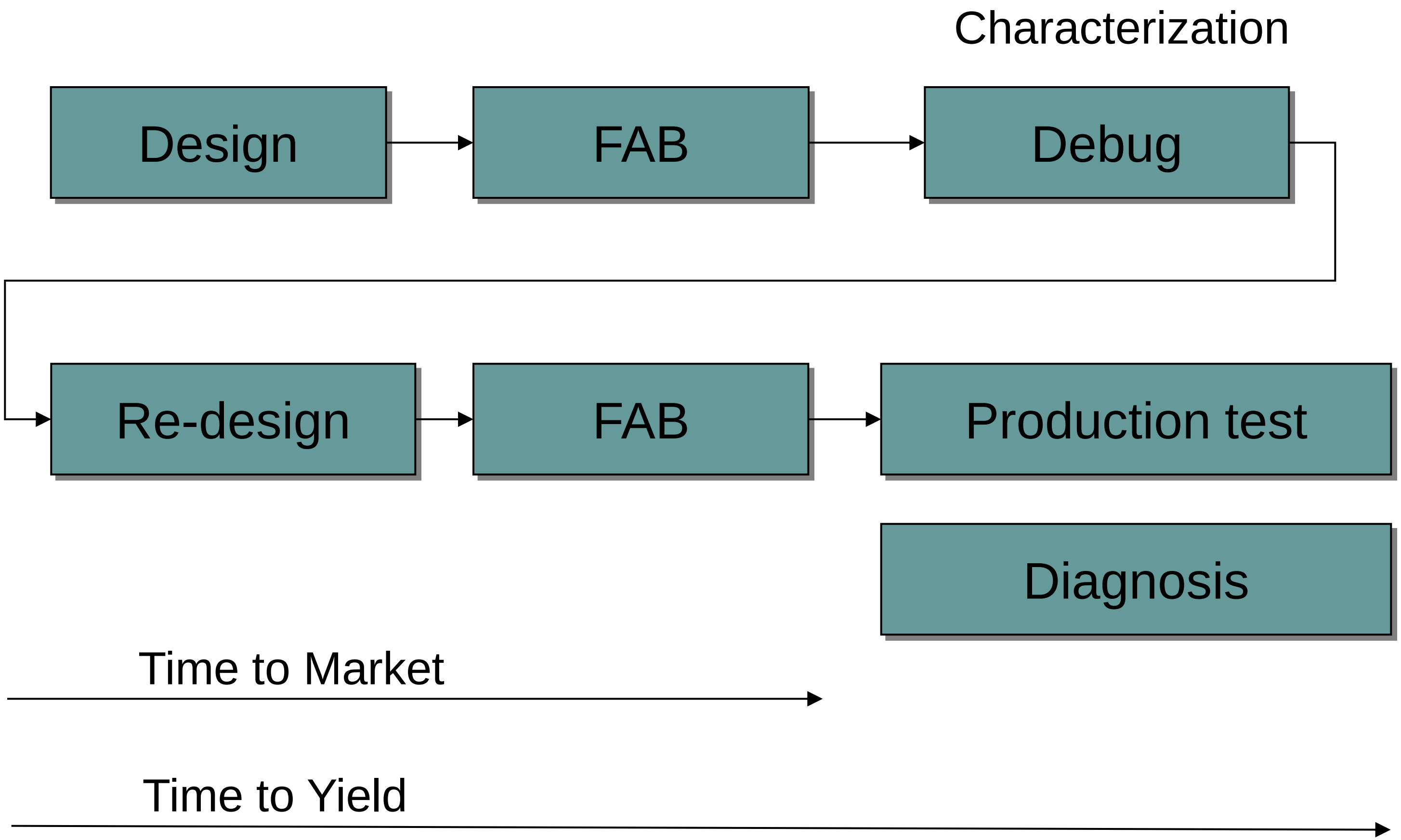
- Failure mode analysis (FMA) is applied to failed parts.
- To locate the cause of misbehavior after the incorrect behavior is detected.
- Results can be used to improve the design or the manufacturing process.
- An important step for improving chip production yield.

Multiple Design Cycles



Long iterations → Late time-to-market/production

A Broad View of Chip Design and Production Phases



What Are We After in Testing?

- Design errors (first silicon debug)
  - Design rule violation
  - Incorrect mapping between levels of design
  - Inconsistent specification
- Manufacturing defects
  - Process faults/variation
  - Time-dependent failures (reliability)
  - Packaging failures

Various Design Errors

Breakdown of design errors in Pentium 4.

- Goof (12.7%) - typos, cut and paste errors, careless coding.
- Miscommunication (11.4%)
- Microarchitecture (9.3%)
- Logic/Microcode change propagation (9.3%)
- Corner cases (8%)
- Power down issues (5.7%) - clock gating.
- Documentation (4.4%)
- Complexity (3.9%)
- Random initialization (3.4%)
- Late definition of features (2.8%)
- Incorrect RTL assertions (2.8%)
- Design mistake (2.6%) - the designer misunderstood the spec

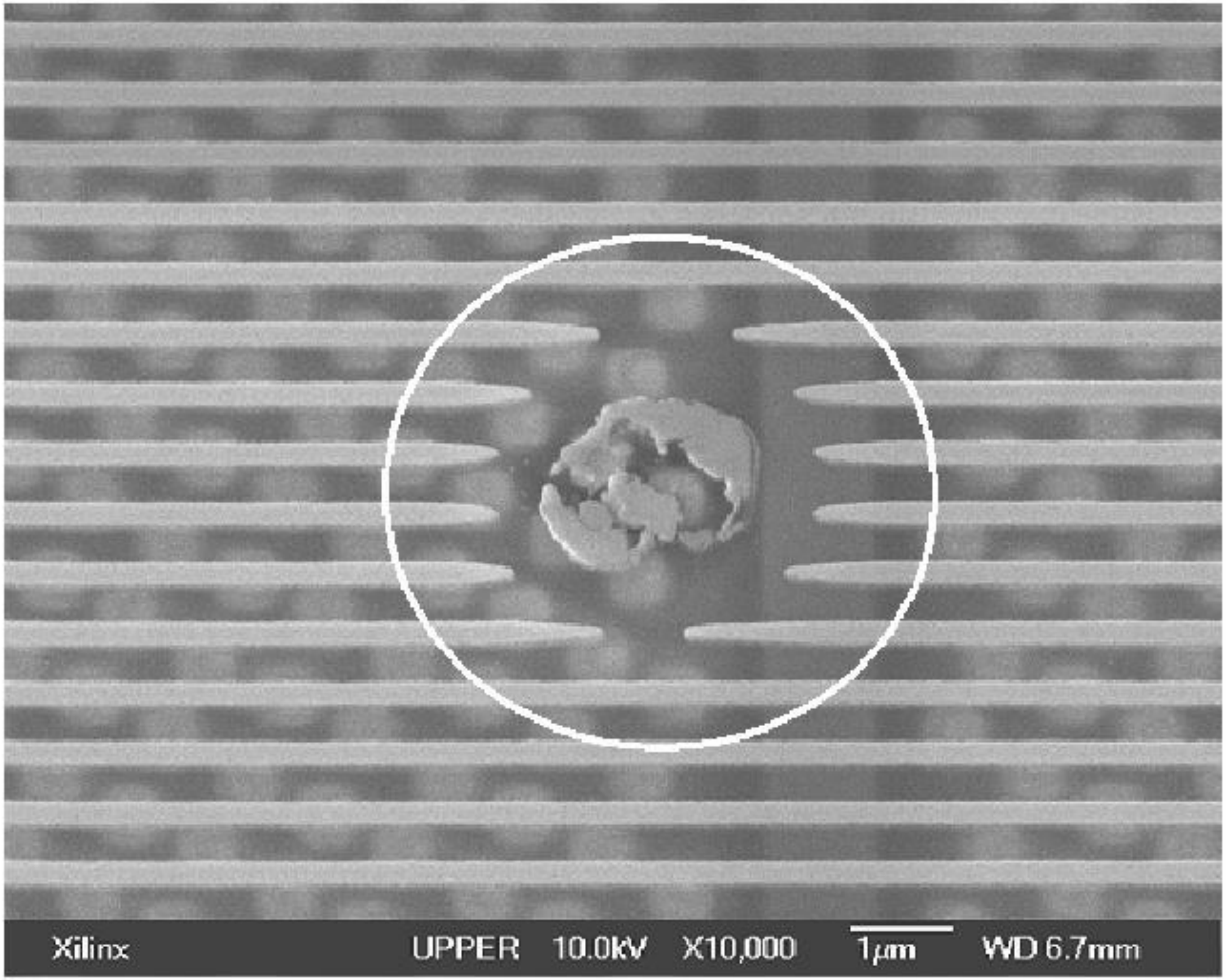
Source: Bentley, DAC2001

Methods to Find First-Silicon Bugs

- Post-silicon debug requires a lot of efforts
  - System Validation (71%).
  - Compatibility Validation (7%)
  - Debug Tools Team (6%)
  - Chipset Validation (5%)
  - Processor Architecture Team (4%)
  - Platform Design Teams and Others (7%)

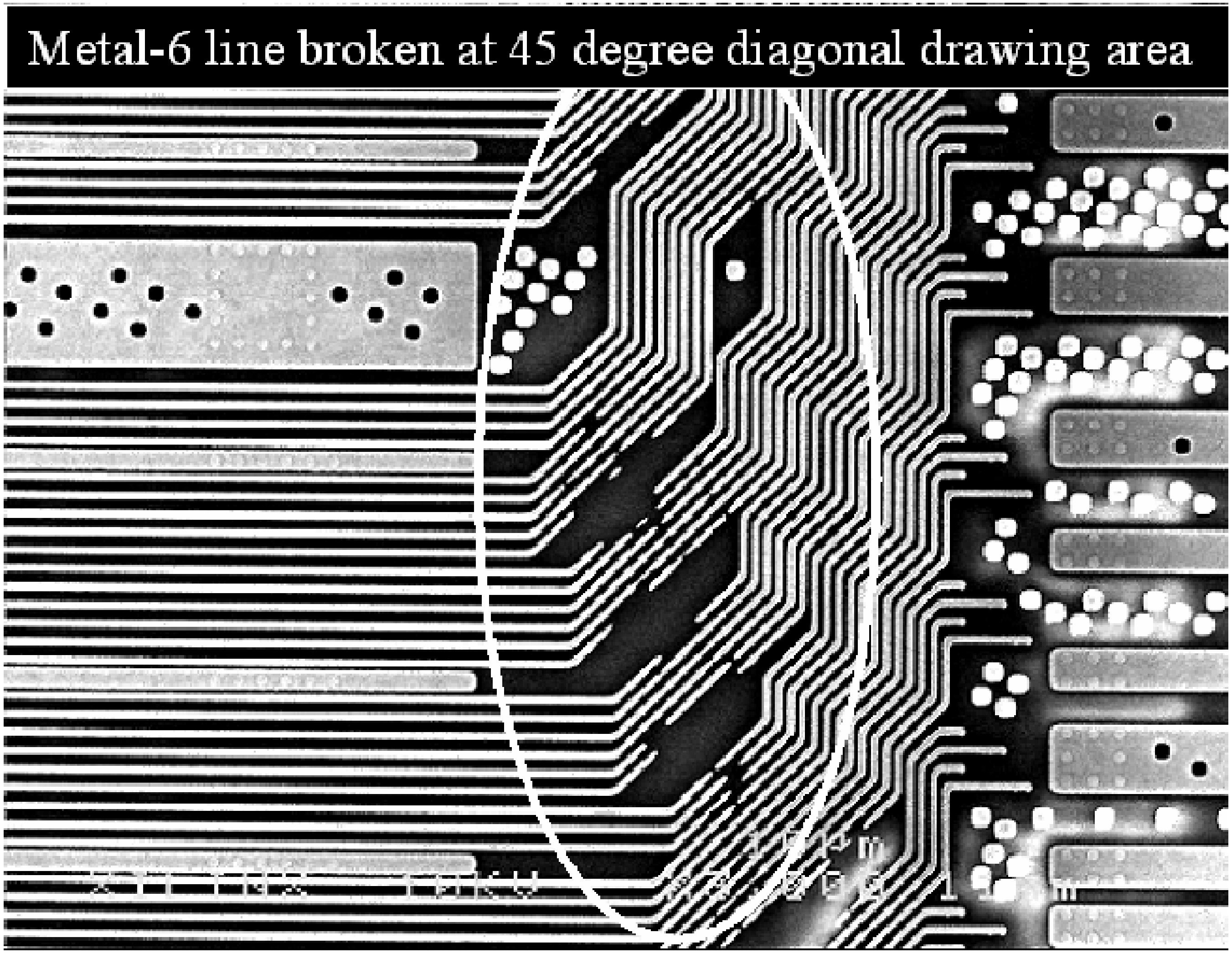
Source: Intel Technology Journal Q1, 2001
Validating The Intel Pentium 4 Processor

Defect Example: Particle



Source: ITC2004, D. Mark J. Fan, Xilinx

Defect Example: Metal breaks



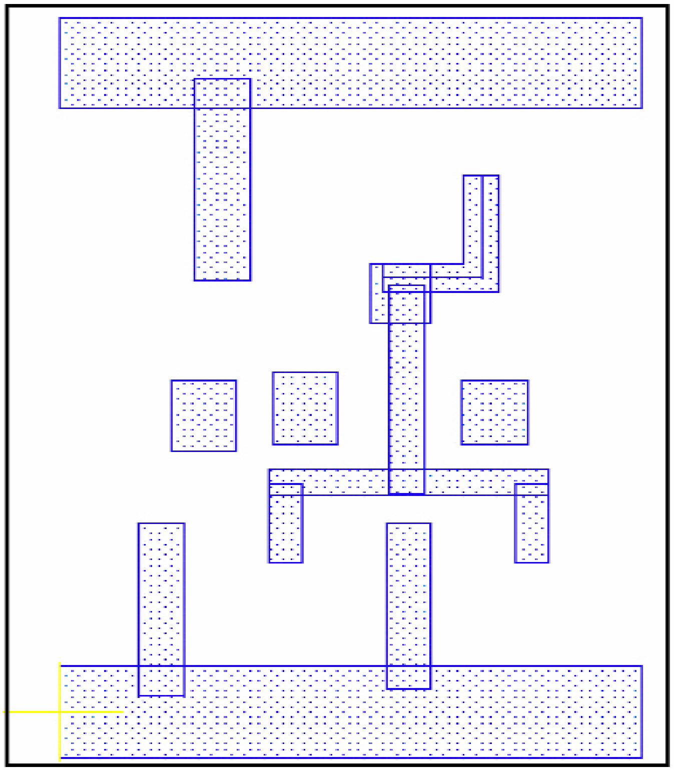
Source: ITC2004, D. Mark J. Fan, Xilinx

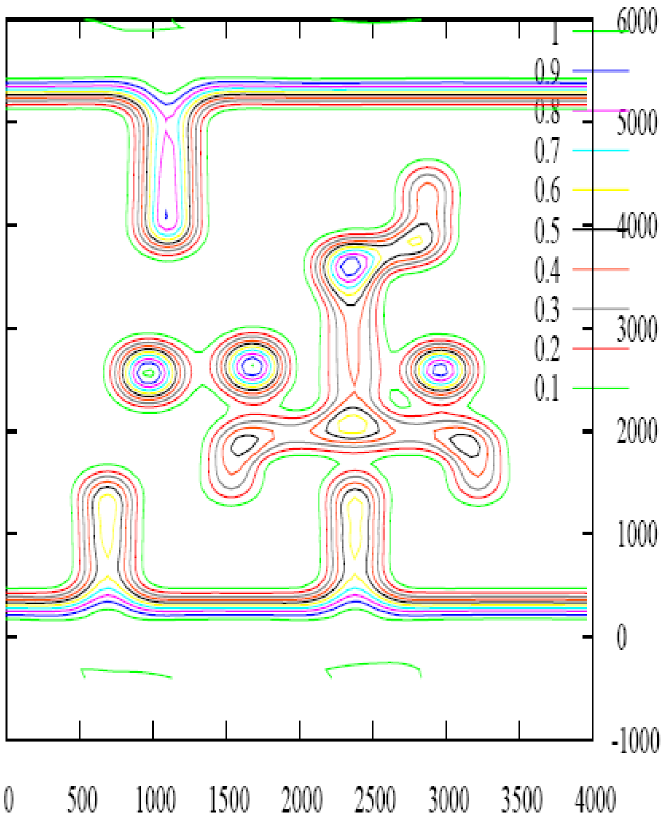
Defect Example: Bridging

Source: ITC1992 Rodriguez-Montanes, R.; Bruis, E.M.J.G.; Figueras, J.

Systematic Process Variations

- Metal layer of NOR3XL standard Cell





Tests Before and After Production

- (Before) Characterization Testing
  - For design debug and verification
  - Usually performed on designs prior to production
  - Verify the correctness of the design & determine exact device limits
  - Comprehensive functional, DC and AC parametric tests
  - Set final spec. and develop production tests
- (After) Production Testing
  - To enforce quality requirements
  - Applied to every fabricated parts
  - Test vectors should be as short as possible under the constraints of test costs and product quality
  - Test costs are the main drivers

Test Items for Production Testing

- Circuit probe test (CP)
  - Examine each part on the wafer before it is broken up into chips
- Final test (FT)
  - Examine each part after packaging
- Usually FT includes
  - Contact test
  - DC parameter test
  - Functional test
    - Make sure circuits function as required by specification.
    - Consume most test resources in production.
- Burn-in test (optional)
  - Exercise chips in extreme conditions, e.g., high temperature or voltage, to screen out infant mortalities
- Speed binning (optional)
  - Determine the max speed of a chip and sell it accordingly

An Exemplary Test Flow

Step		
CP	Objective:	gross process defect
	Metric:	coverage of targeted faults
	Patterns:	functional / scan / BIST
FT	Objective:	process defect, package defect
	Metric:	coverage of targeted faults
	Patterns:	functional / scan / BIST
Burn-in	Objective:	aging defects
	Metric:	toggle coverage
	Patterns:	functional / scan (without comparison)
Speed binning	Objective:	performance
	Metric:	speed, delay fault coverage
	Patterns:	functional (mostly) / scan (rare)
Quality Assurance test	Objective:	Final quality screen
	Metric:	Adhoc
	Patterns:	Functional, System

Connectivity Test

- Verify whether the chip pins have opens or shorts
- Also called open/short test
- Draw current out of the device and measure voltage at the input pin
- Utilize the forward bias current of the protection diodes at the pin to determine whether a short or open exists

DC Parametric Test

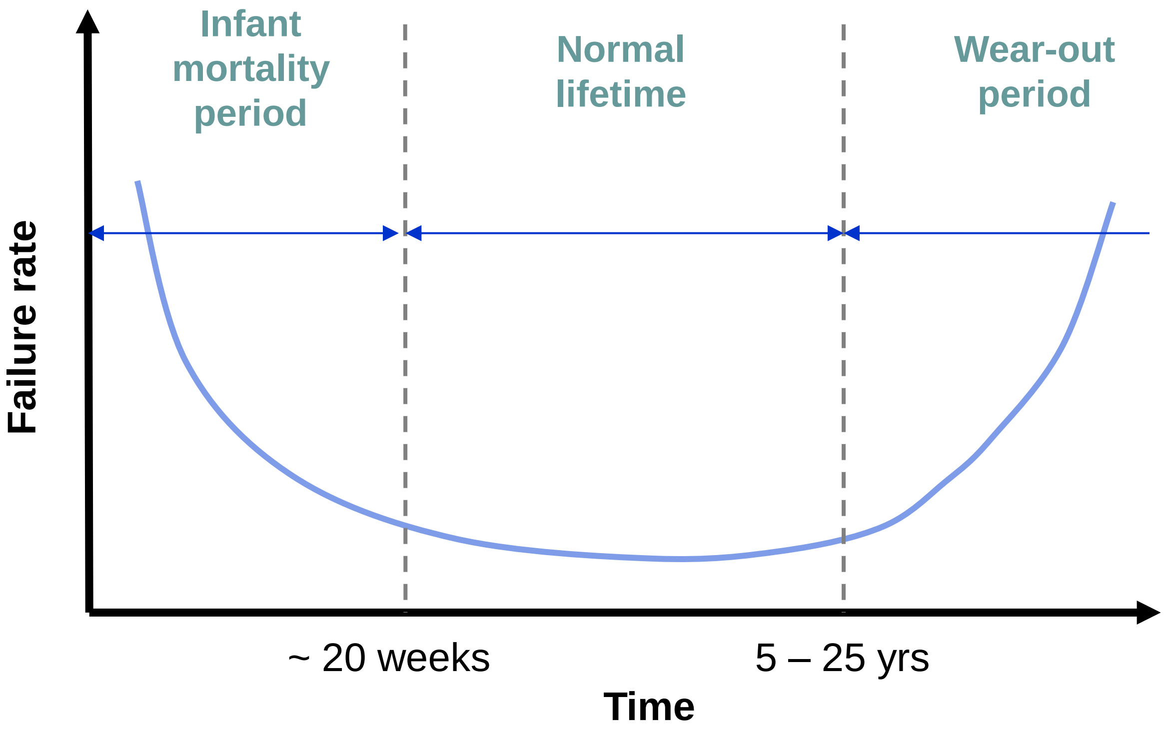
- Tests performed by Parametric Measurement Unit (PMU)
- Much slower than the normal operation speed
  - Static (operating) current test
    - check the power consumption at standby (operating) mode
  - Output short current test
    - Verify that the output current drive is sustained at high and low output voltage
  - Output drive current test
    - For a specified output drive current, verify that the output voltage is maintained

AC Parametric Test

- To ensure that value/state changes occur at the right time
- Some AC parametric tests are mainly for characterization and not for production test.
  - Test for rise and fall times of an output signal
  - Tests for setup, hold and release times
  - Tests for measuring delay times
    - E.g. tests for memory access time

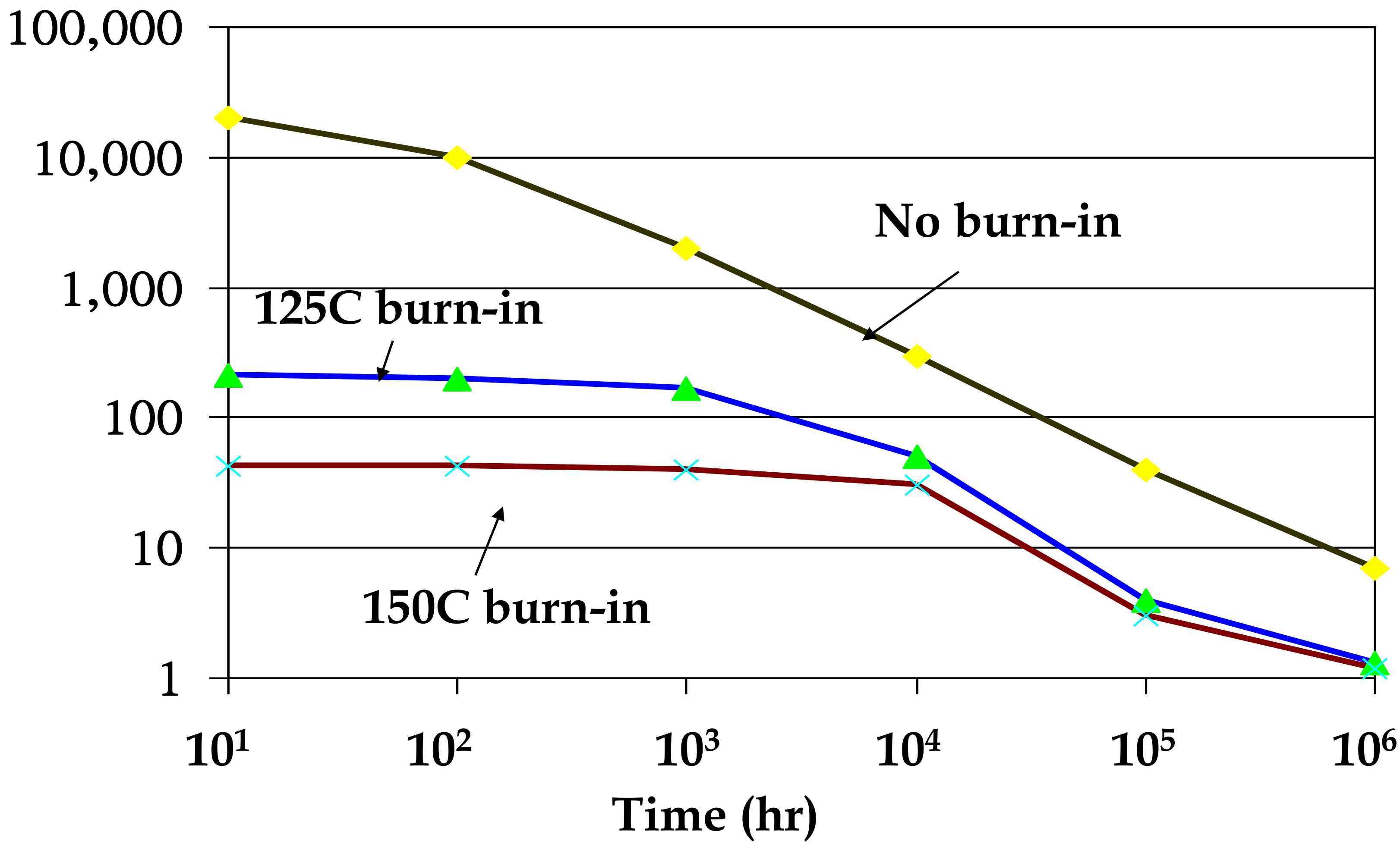
Burn-in Test

- Early failure detection reduces cost
  - Burn-in to isolate infant mortality failures



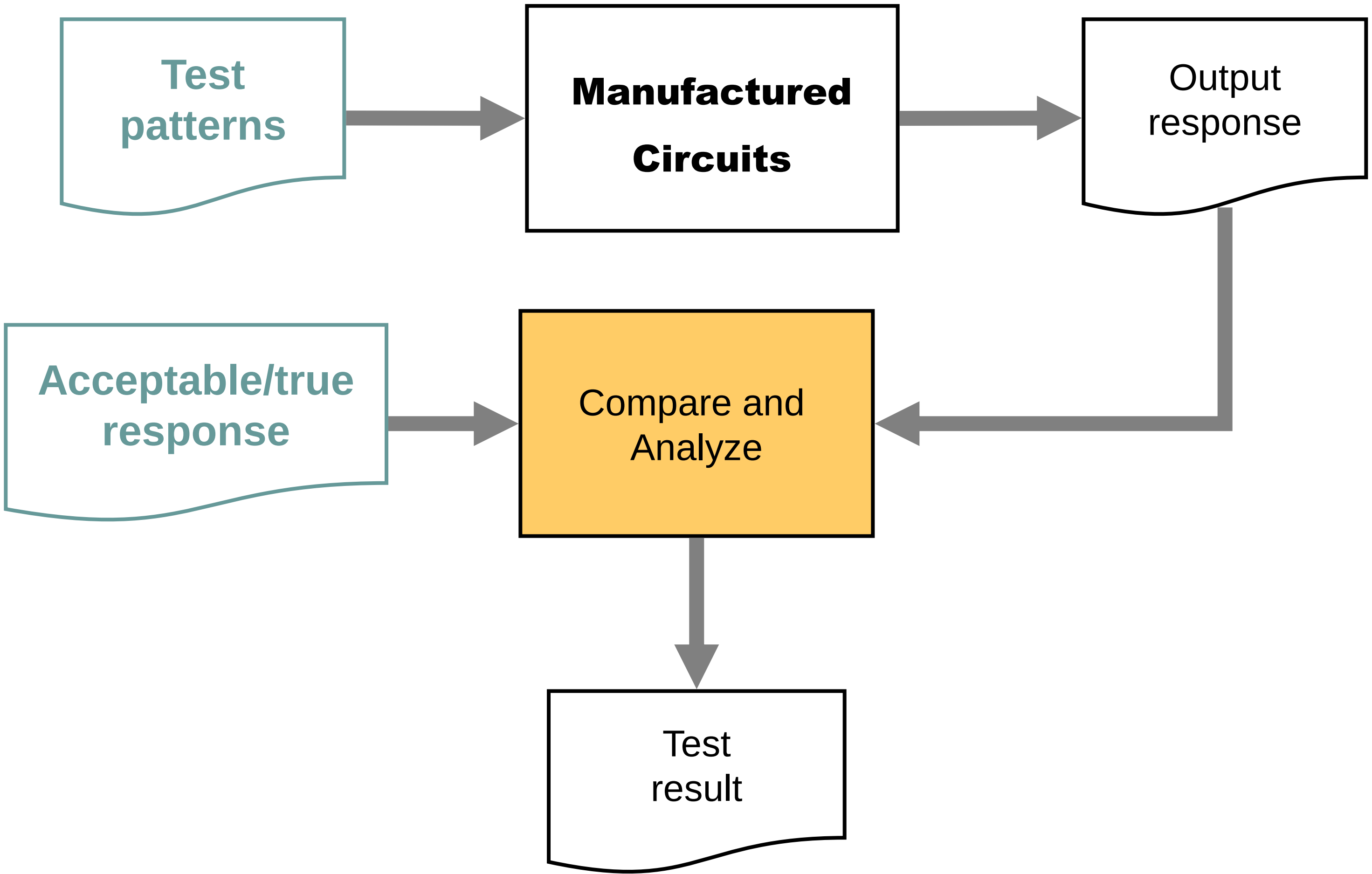
Bathtub Curve of IC's Failure Rate

An Example of IC Failure Rate vs. System Operating Time With/Without Burn-in

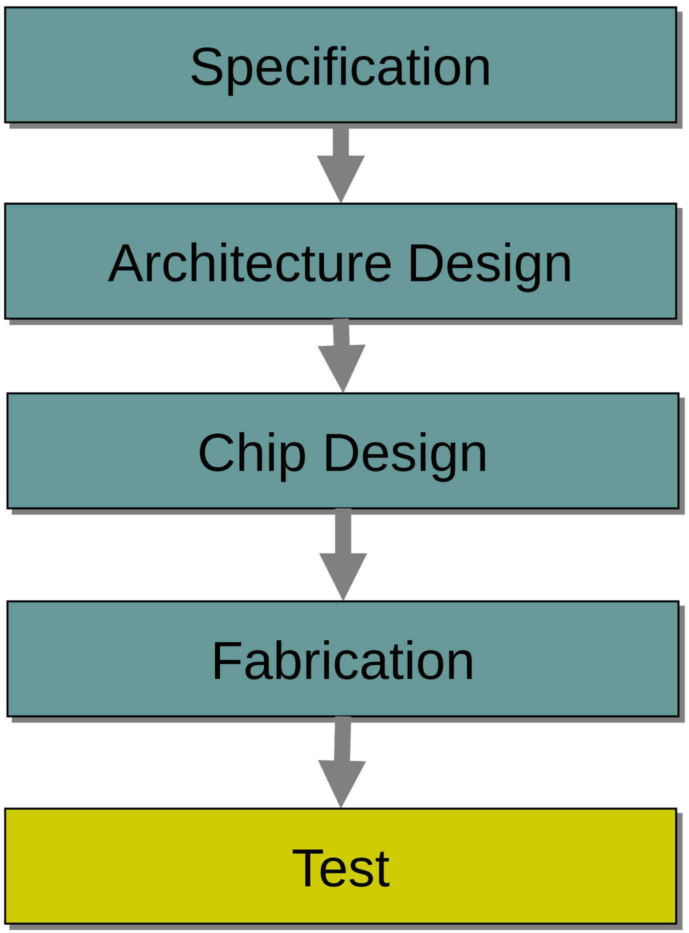


Functional Test

- Selected test patterns are applied to circuits and response are analyzed for functional correctness.



Activities for Developing Functional Test



- Generate test pattern
- Evaluate the quality of test patterns
- Design circuit with better test efficiency

- Apply test patterns

Key Issues of Functional Test

- Where does patterns come from?
  - Design simulation patterns (Functional patterns)
  - Automatic test pattern generation (ATPG)
- How to evaluate the quality of test patterns?
  - Fault coverage evaluation
- How to improve test efficiency?
  - Design for Testability (DFT)
- How to apply test patterns?
  - Automatic test equipments (ATE)
  - Built-in self test (BIST)

Functional v.s. Structural Test

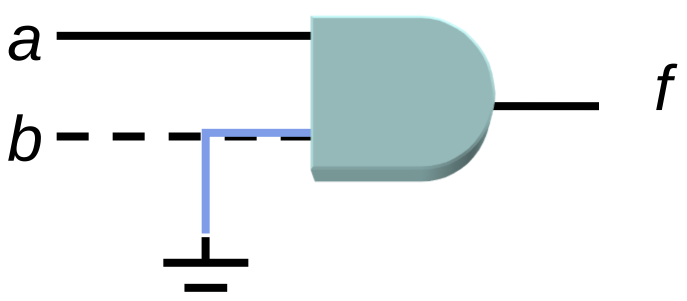
- Functional test
  - Exercise the functions according to the spec
  - Often require designers' inputs
  - Large number of patterns with low fault coverage
  - Difficult to be optimized for production tests
- Structural test
  - Use the information of interconnected components (e.g., gates) to derived test regardless of the functions
  - Fault modeling is the key
  - Basis of current testing framework---ATPG, Fault simulator, DFT tools, etc.

Fault Models

- Fault modeling is a way to represent the cause of circuit failure.
- Model the effects of physical defects by the logic function and timing
  - Enumeration of real defects is impossible
- Makes effectiveness measurable by experiments
  - Fault coverage can be computed for specific test patterns to reflect its effectiveness

Single Stuck-At Fault Model

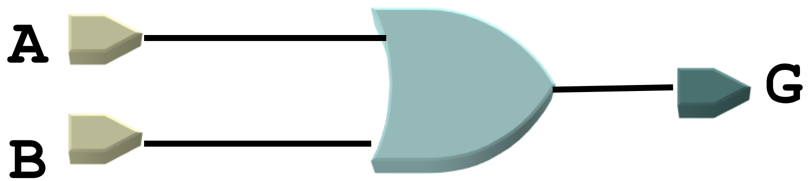
- Assumptions:
- Only One line is faulty
- Faulty line permanently set to 0 or 1
- Fault can be at an input or output of a gate



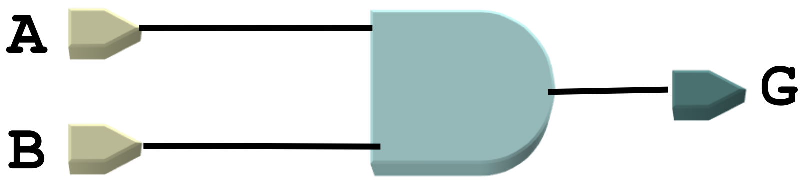
- One of the gate input terminal was mistakenly connected to ground
- Fault: *b* stuck at 0
- signal b will always be "0"

Logic Gate Basics

OR Gate



AND Gate

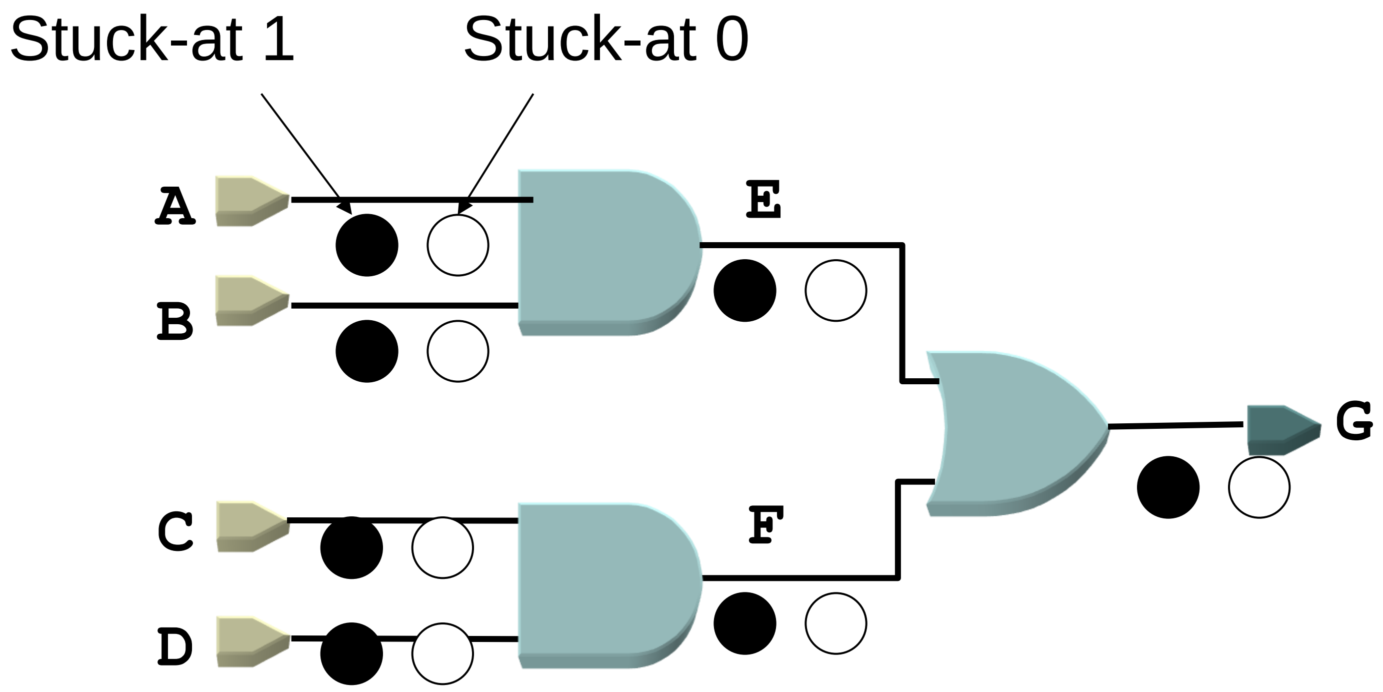


A	B	G
0	0	0
0	1	1
1	0	1
1	1	1

A	B	G
0	0	0
0	1	0
1	0	0
1	1	1

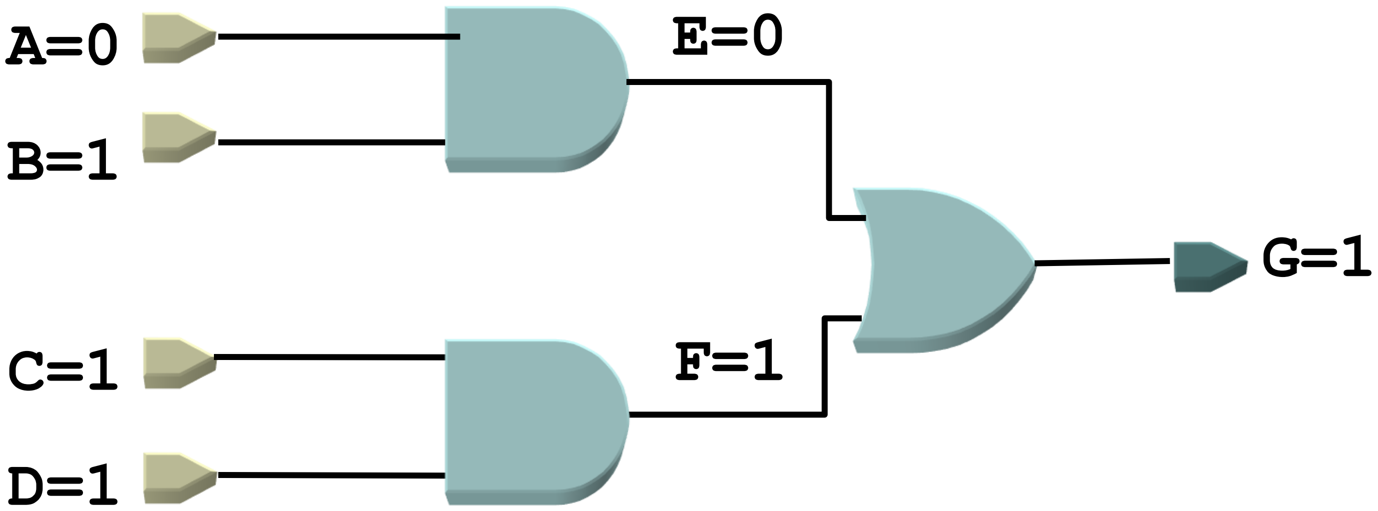
Only binary values, 0 and 1, will be used.
A and B are inputs and G is the output.

Stuck-At Faults Example



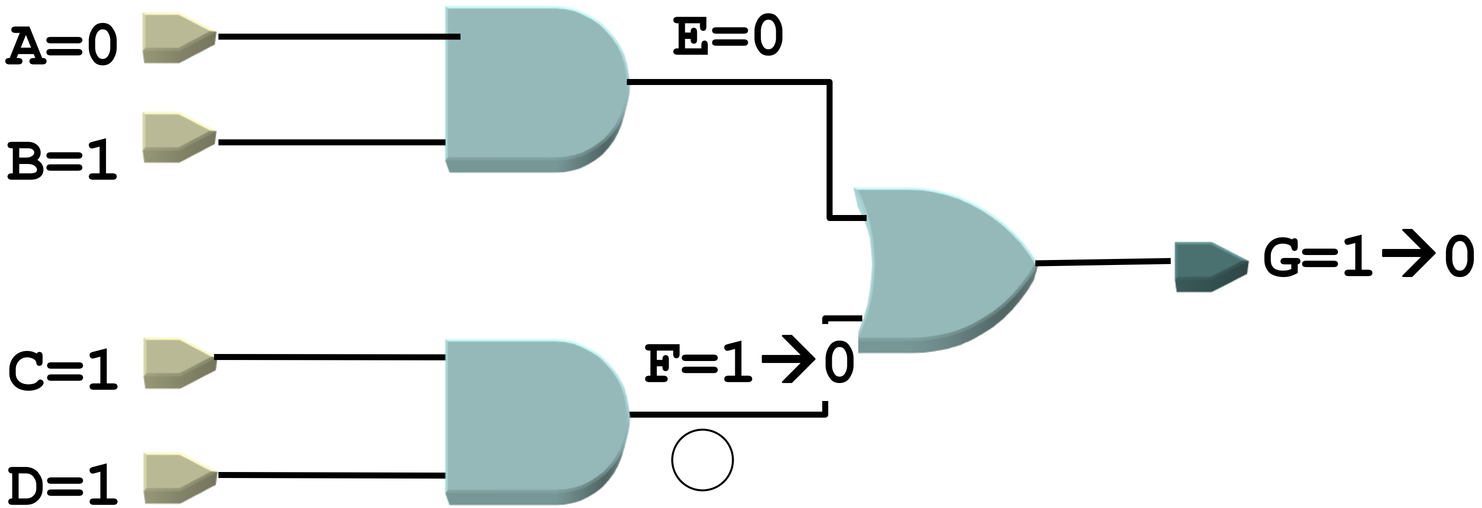
Total Faults = *Nf = 2* total number of signals = 2* 7=14*

A Simple Simulation with Input (ABCD)=(0111)



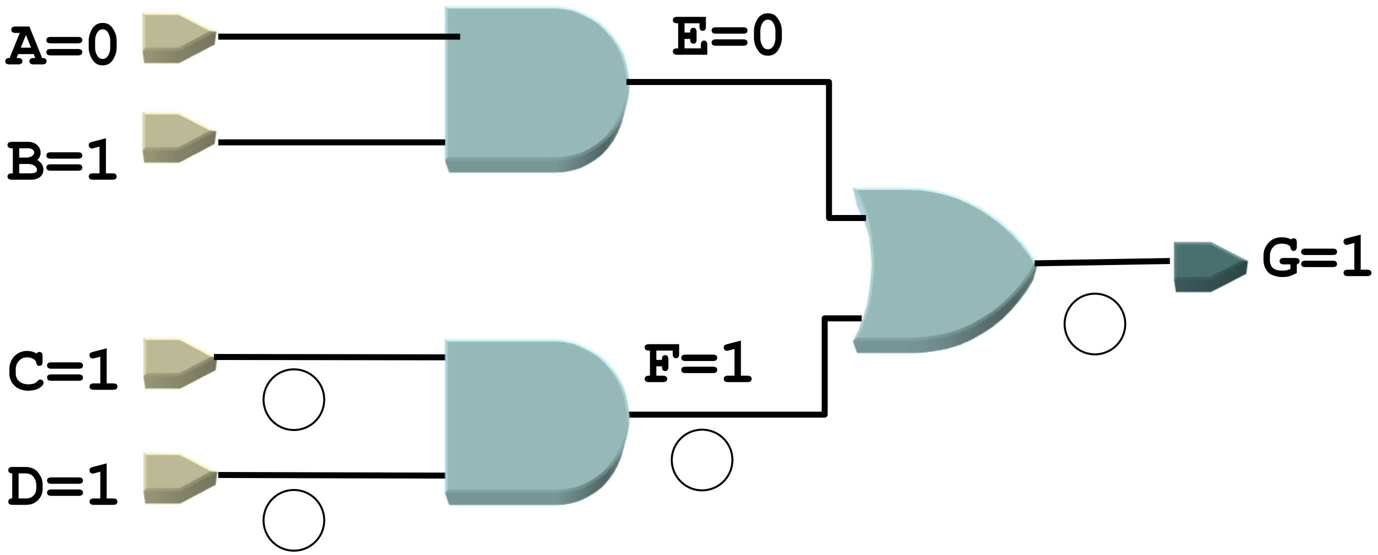
We use logic simulation to propagate (transfer) input values to outputs.

What if F stuck-at-0 occurs with (ABCD)=(0111)



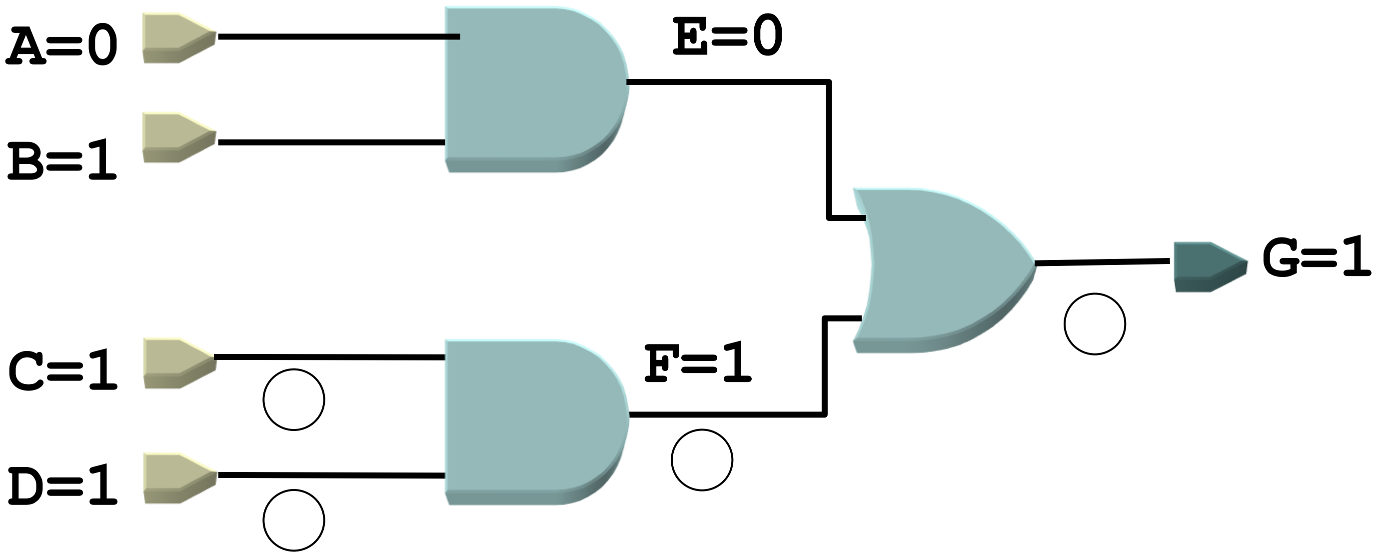
We use logic simulation to propagate (transfer) faulty values to outputs.
For this case, we say (0111) covers the fault F stuck-at-0.

Other Faults Covered By (ABCD)=(0111)



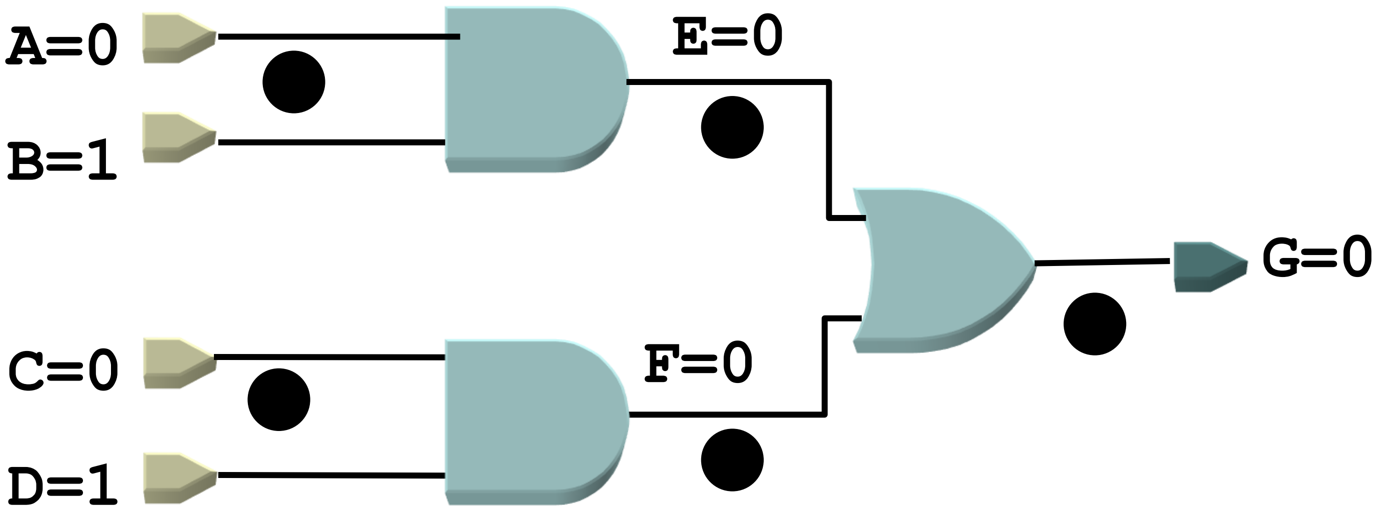
By performing several logic simulation with faults (**fault simulation**), we found (0111) covers four faults: C, D, F, and G stuck-at-0.

Fault Coverage of (ABCD)=(0111)



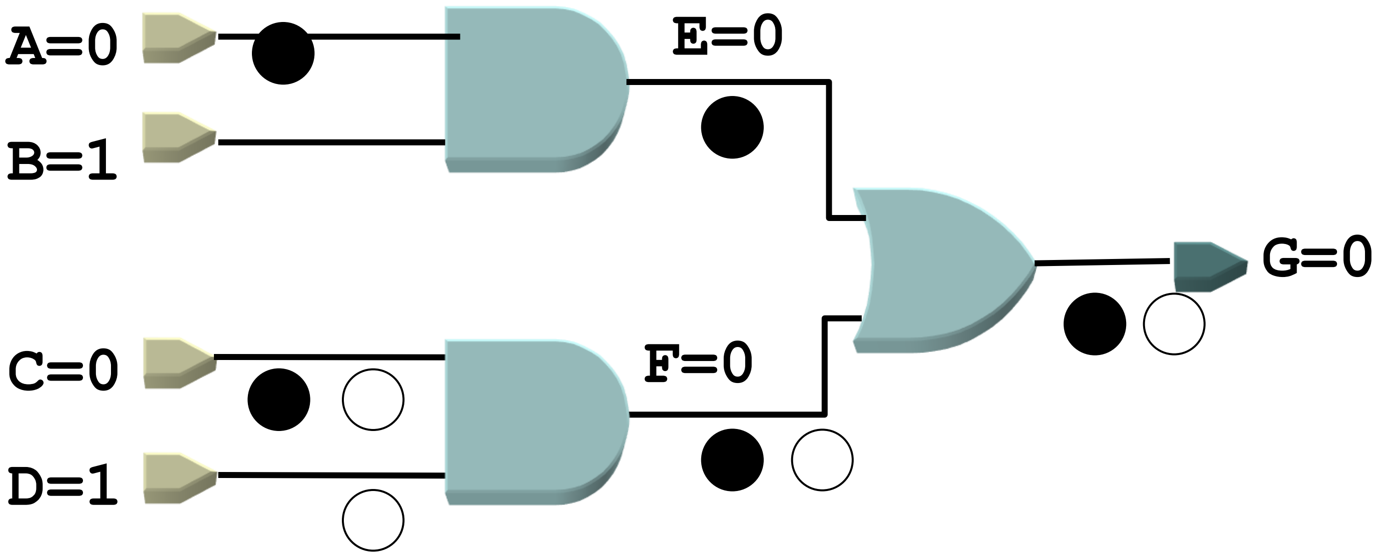
Since (0111) covers four faults: C, D, F, and G stuck-at-0.
And total number of faults is 14.
We say (0111) has a fault coverage of 4/14 ~ 28.6%

Fault Coverage of (ABCD)=(0101)



Since (0101) covers four faults: A, C, E, F, and G stuck-at-1.
And total number of faults is 14.
We say (0101) has a fault coverage of 5/14 ~ 35.7%

Combined Fault Coverage of (ABCD)=(0111) and (0101)



We know that both vectors cover different faults, so the total number of covered faults are 4+5.
Therefore we have a total fault coverage 9/14 ~ 64.3%

Fault Coverage

- Fault Coverage T
  - Is the measure of the ability of a set of tests to detect a given class of faults that may occur on the device under test (DUT)

$$T = \frac{\text{No. of detected faults}}{\text{No. of all possible faults}}$$

- Fault simulation is used to evaluate fault coverage for test patterns.

Meaning of Fault Coverage

- Our goal in testing is to find **test patterns** to achieve **100% fault coverage.**
- Under the assumption of the fault model (e.g., single stuck-at fault), we've done a good job!
  - Remember the problem of testing a circuit with 50 inputs?
  - Remember the problem of numerous defects that can occur in a chip?
- Though single stuck-at fault model is very simple, it is very effective.
  - Other fault models is still needed to further improve chip quality.

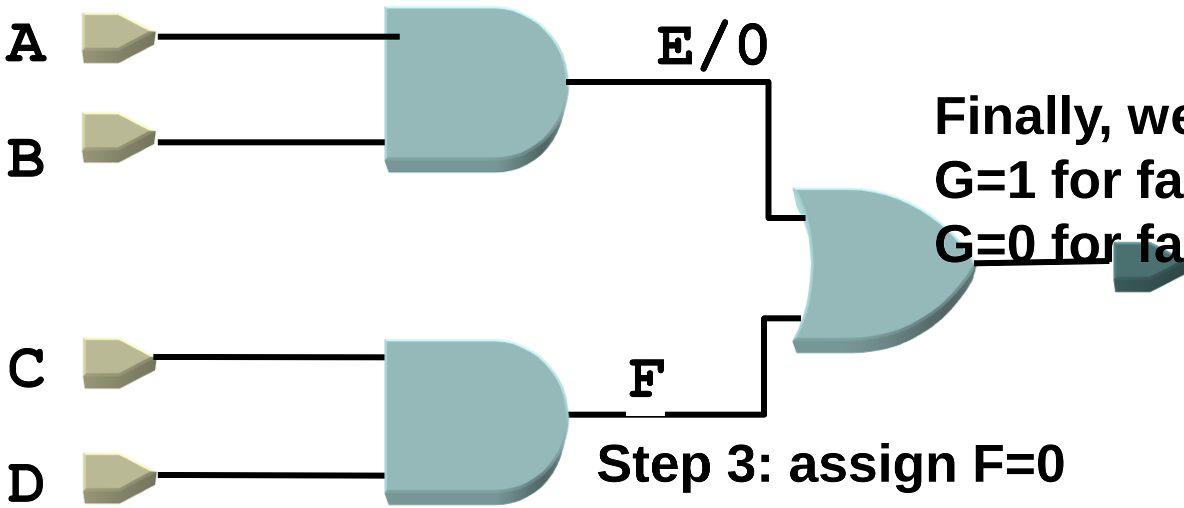
Automatic Test Pattern Generation (ATPG)

- Generate test patterns to cover modeled faults automatically.
- A complex process to determine the quality of tests
  - The most time-consuming process in test development
- Very difficult for sequential circuits (circuits has memory elements).

An Example of ATPG for E stuck-at-0

Step 2:
assign A=1 and B=1

Step 1: assign E=1



Finally, we will see
G=1 for fault-free circuits, and
G=0 for faulty circuits.

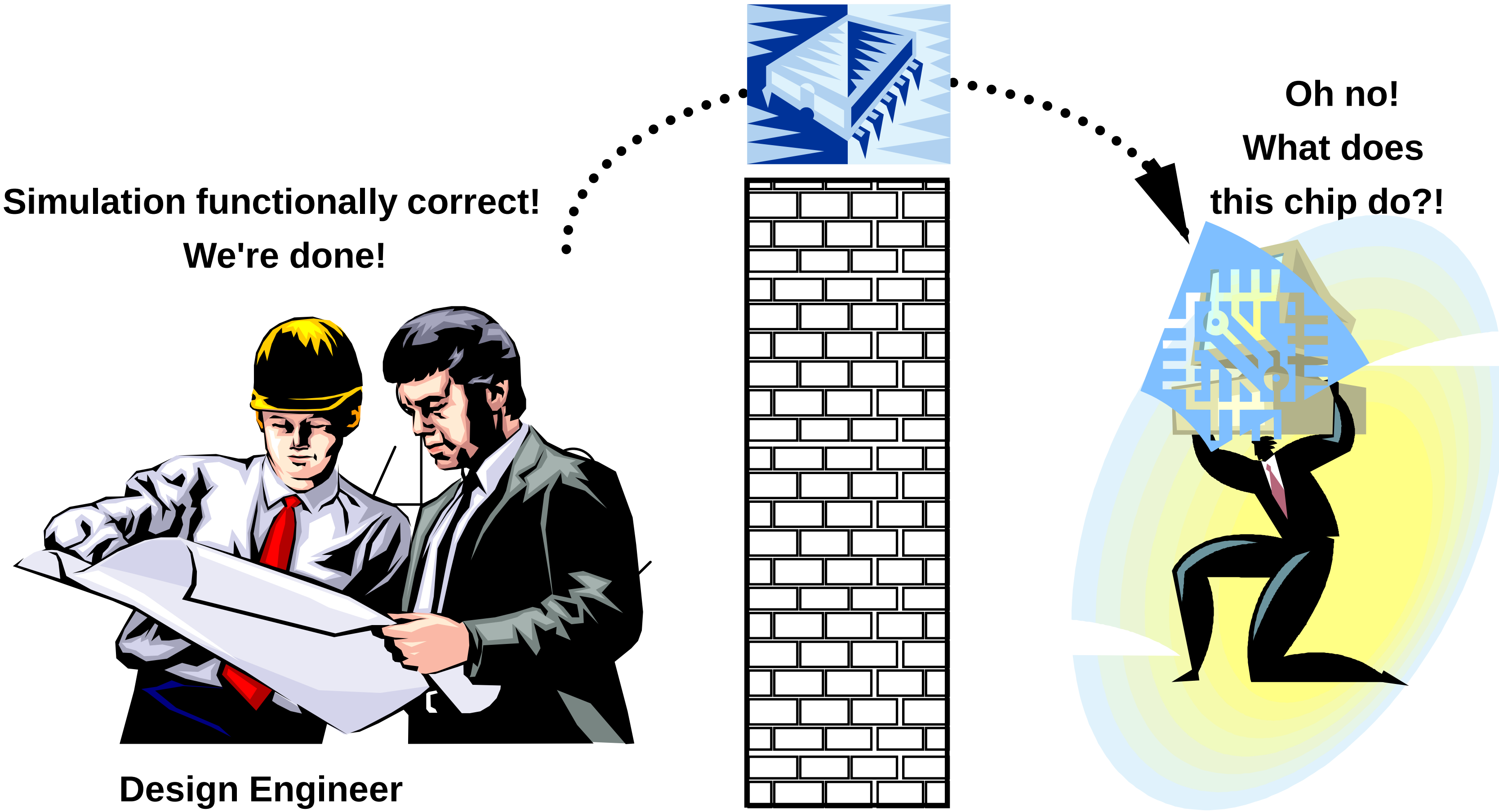
Step 3: assign F=0

Step 4:
assign (C, D)=(0, 0), (0, 1), or (1, 0)

We can have test vectors (A, B, C, D)=(1, 1, 0, 0), (1, 1, 0, 1), (1, 1, 1, 0)

The Infamous Design/Test Wall

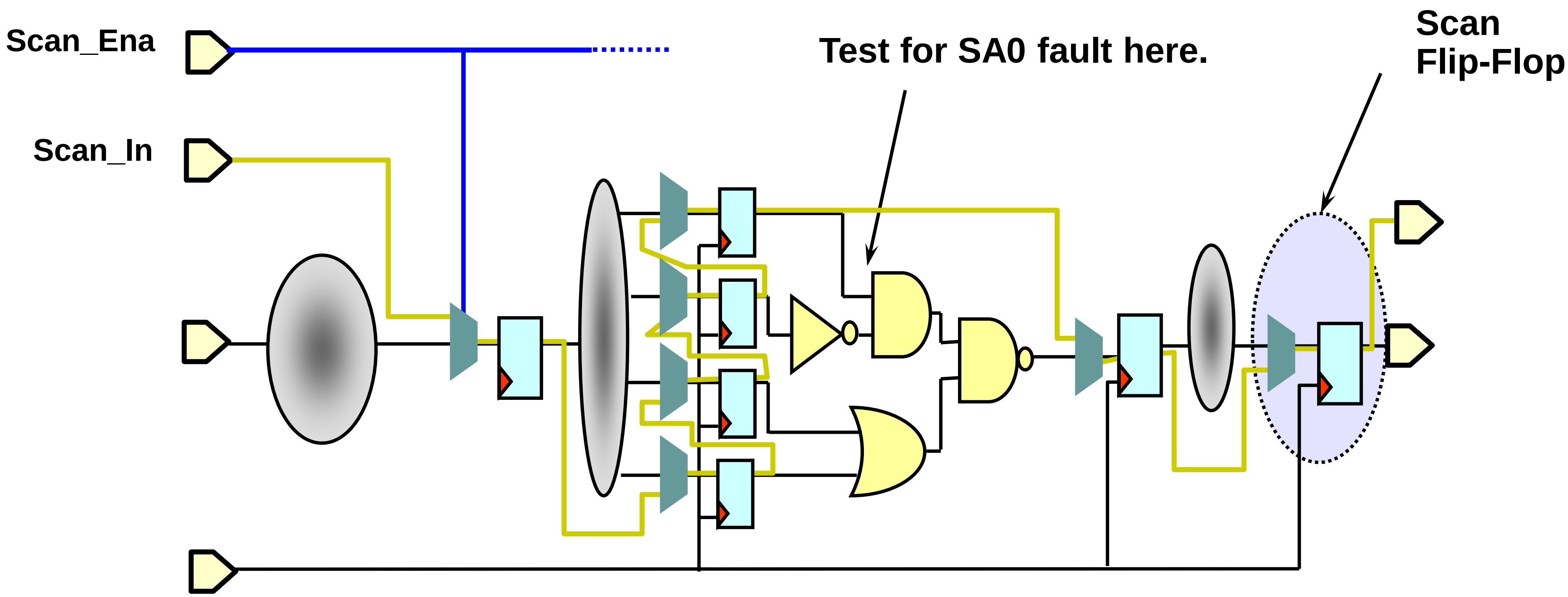
30 years of experience proves that test after design does not work!



Design for Testability (DFT)

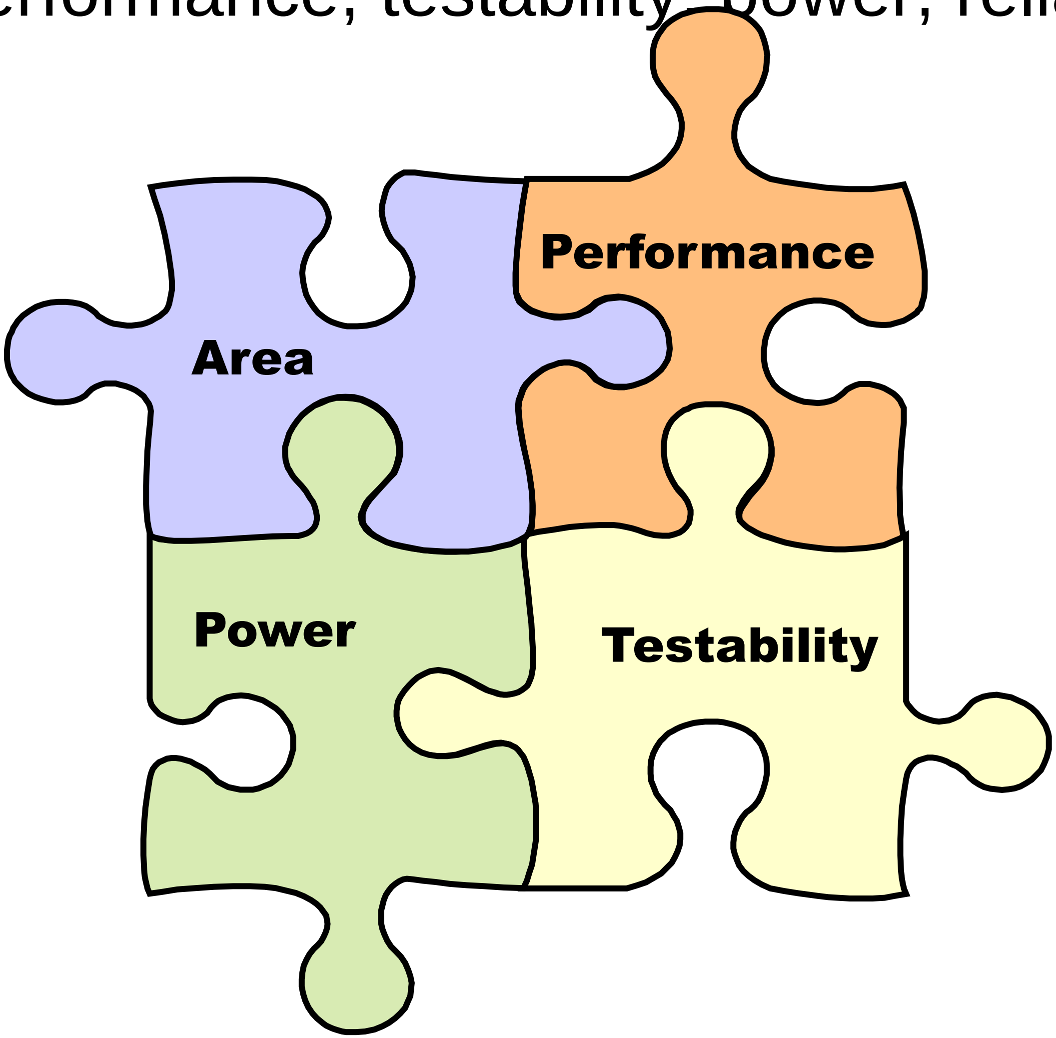
- DFT is a technique to design a circuit to be easily testable
- Add the cost of area/performance, but dramatically reduce cost for tests
- For example, use scan technique to make test generation feasible on sequential circuit.
- A very important step in circuit design to make sure a circuit is testable.

Full Scanned Sequential Logic ---An Example of DfT



Multiple Design Missions

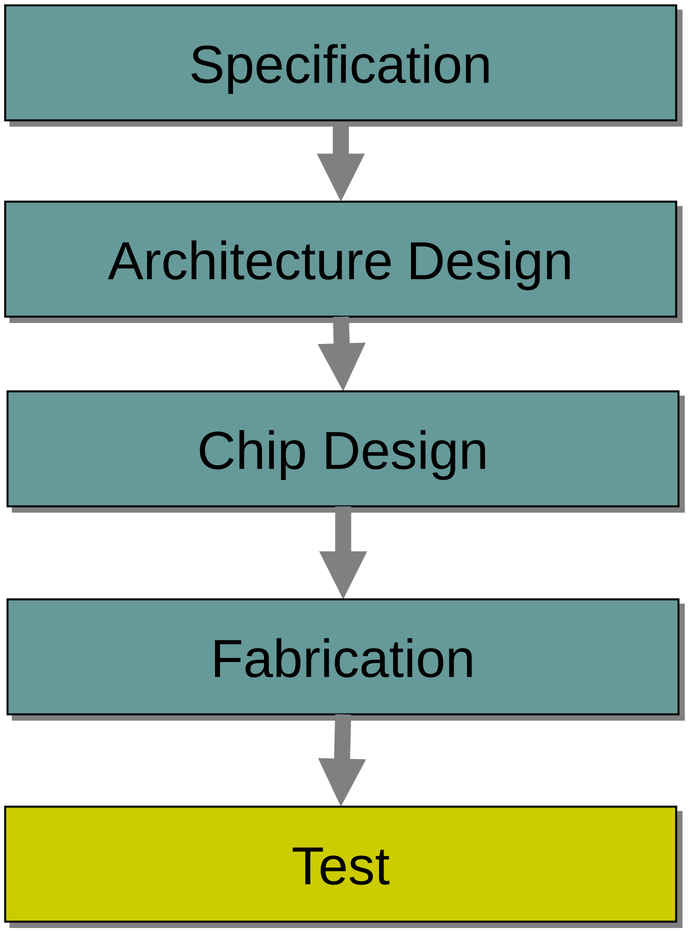
- Chips have to optimally satisfy many constraints: area, performance, testability, power, reliability, etc.



Definition of BIST

- BIST is a DFT technique in which testing (test generation , test application) is accomplished through built-in hardware features.
- Advantages
  - Better quality
  - Reduce test application time
  - Reduce test development time
- Costs
  - Area increased
  - Circuit performance degrade
  - Yield loss

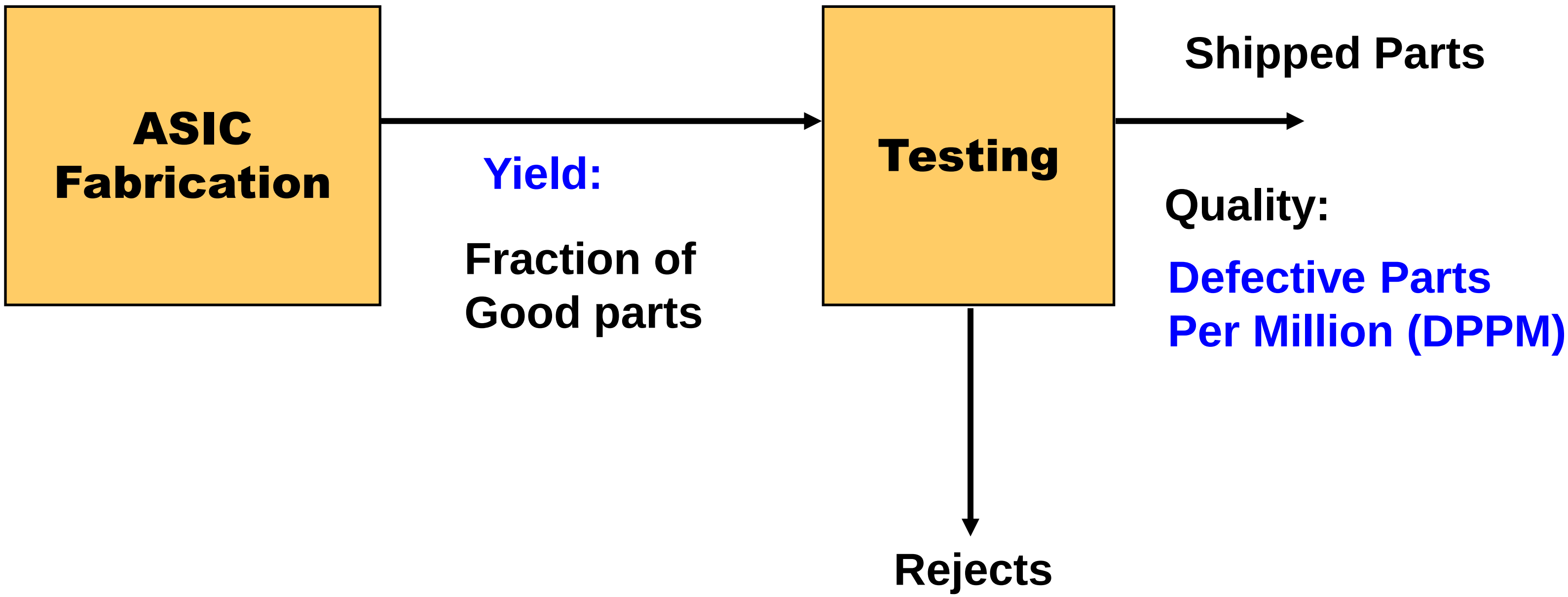
Tools for Developing Functional Tests (Recap)



- DFT
- BIST
- ATPG
- Fault simulation

- ATE
- BIST

Testing and Quality



Quality of shipped part is a function of yield Y and the test (fault) coverage T.

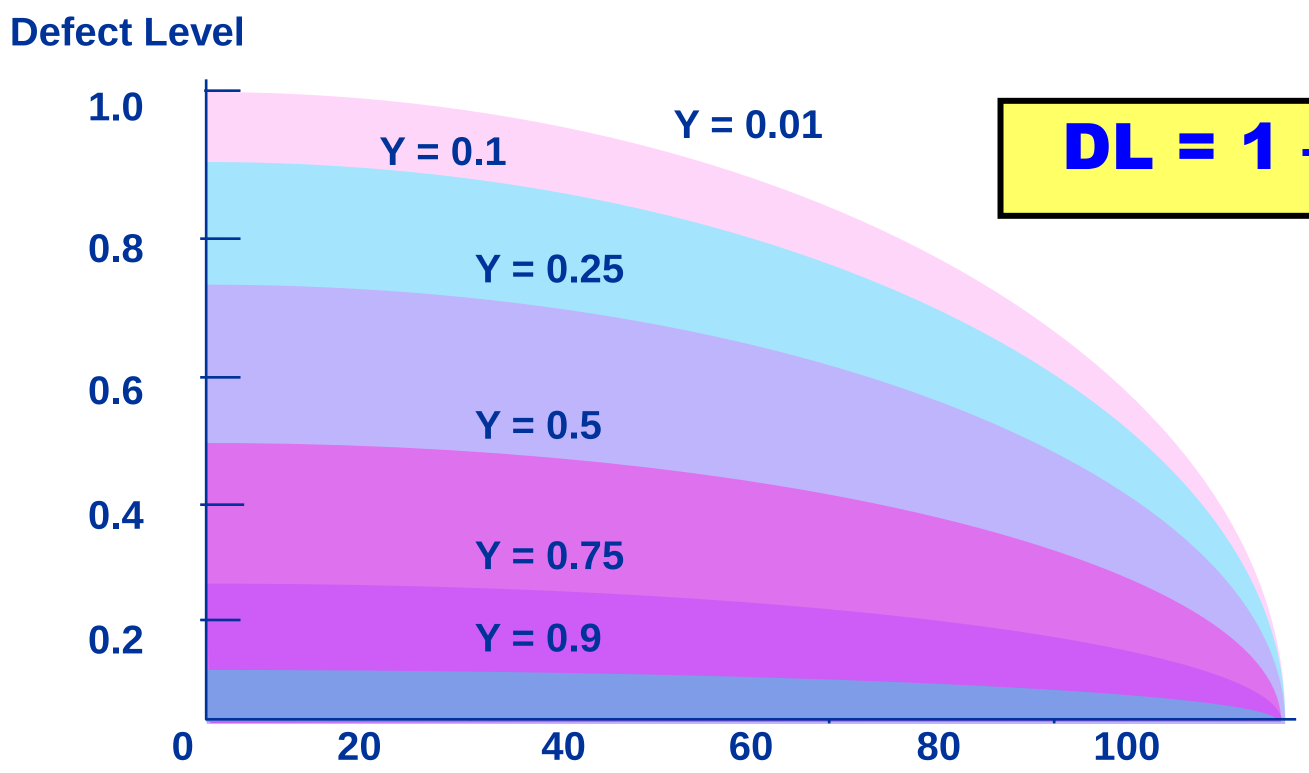
Defect Level

- Defect Level
  - Is the fraction of the shipped parts that are defective

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

Y: yield
T: fault coverage

Defect Level v.s. Fault Coverage



(Williams IBM 1980)

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

High fault coverage ⟶ Low defect level

DPM v.s. Yield and Coverage

Yield	Fault Coverage	DPM
50%	90%	67,000
75%	90%	28,000
90%	90%	10,000
95%	90%	5,000
99%	90%	1,000
90%	90%	10,000
90%	95%	5,000
90%	99%	1,000
90%	99.9%	100

A chip with 100 DPM or below is considered of high quality.

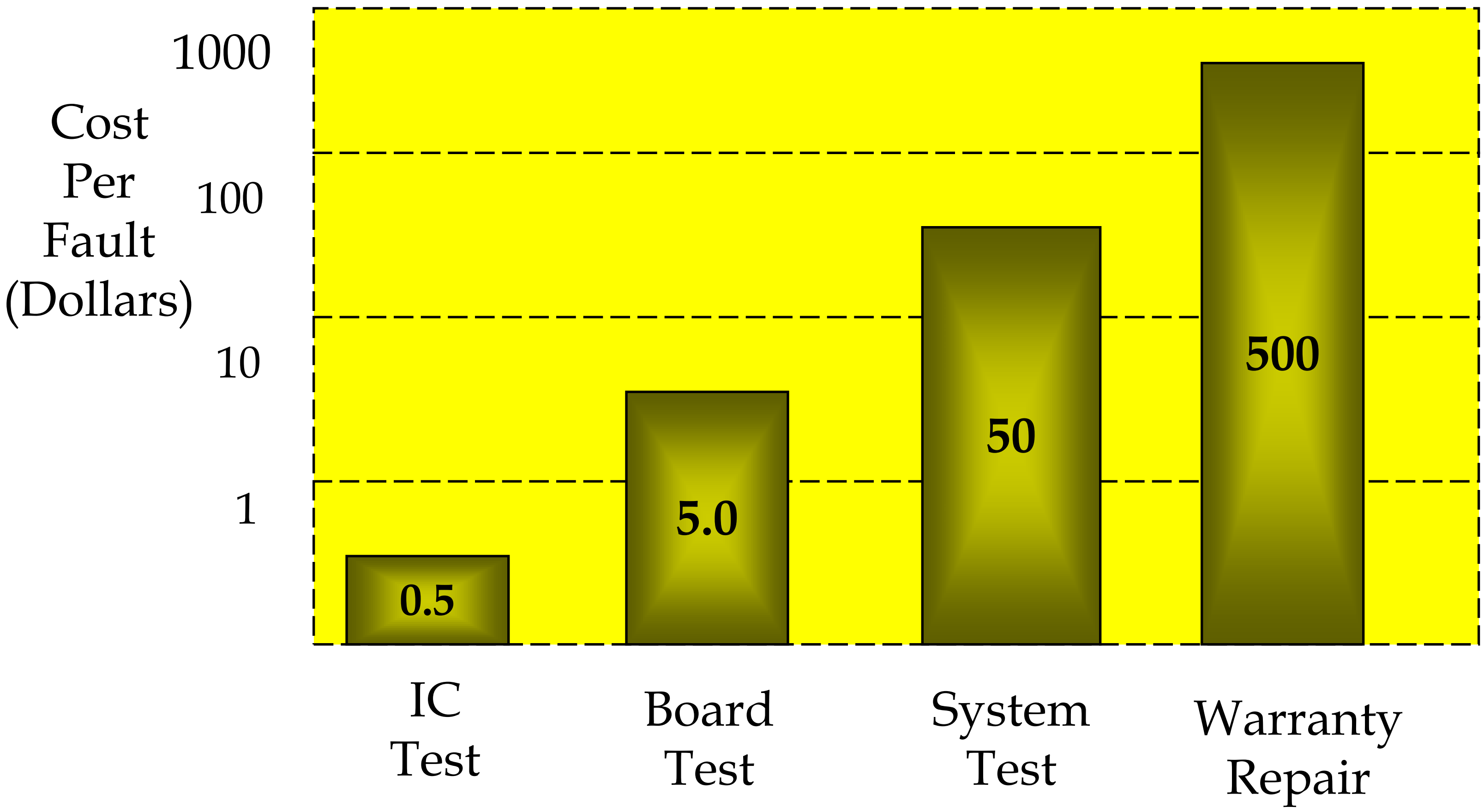
Components of Test Costs (I)

- Determining the costs in each design phase is very important for evaluating different test strategies
- Cost directly impacted by tests
  - Test equipment
  - Test development
    - Test planning, test program development
  - Test time
    - Time using the equipment to support testing
  - Test personnel

Components of Test Costs (II)

- Other costs associated with tests
  - Design time
  - Chip area (manufacturing costs)
  - Time to Market
  - Product quality
    - Impact a company’s image and sales

Cost Of Testing - The Rule of Tens

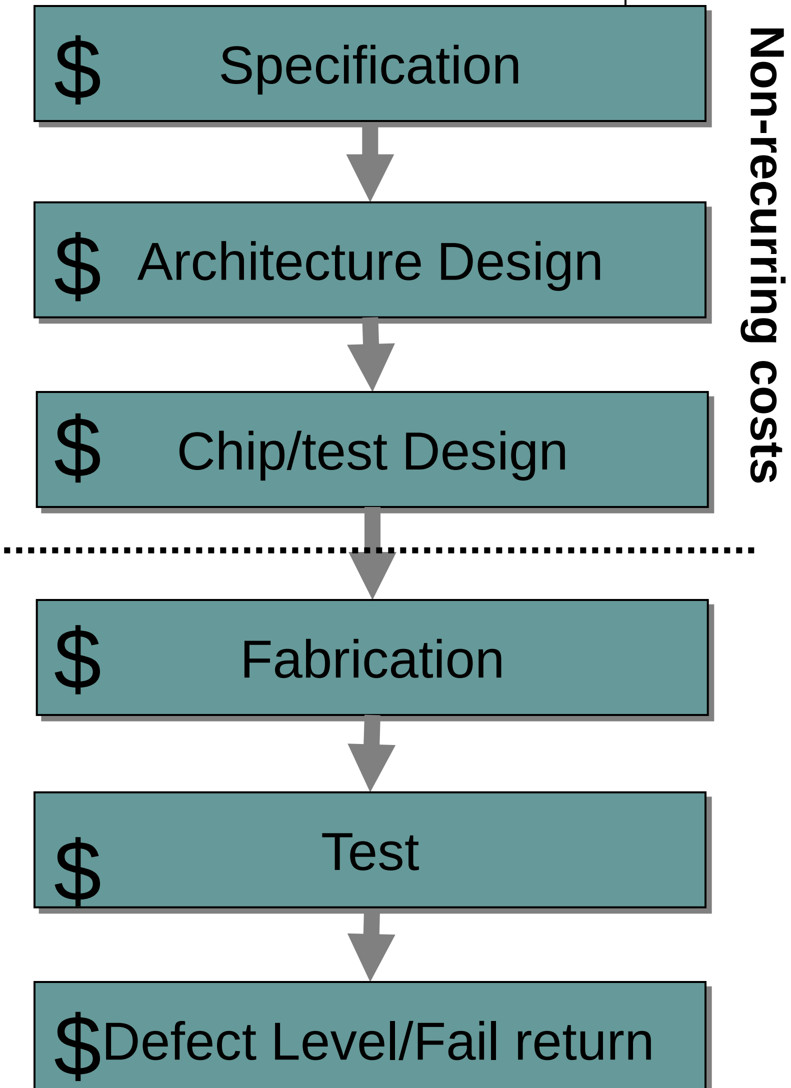


Implications of Rule of Tens

- Early detection can prevent costly diagnosis and replacement later.
- For example, if a bad IC is not detected, the cost to find a board including the bad IC is at least 10 times higher.

Test Economics

- Build an appropriate cost/benefits model based on empirical data of the manufacturing process.
- Evaluate test strategies (DFT; BIST) according to the model
- Customize the model for each project
- Follow and review the model closely through careful management



A Case Study for Test Economics

- **A BIST and Boundary-Scan Economics Framework by JOSÉ M. MIRANDA**
  - Lucent Technologies Bell Laboratories
  - IEEE Design and Test of Computers, JULY–SEPTEMBER 1997

Conclusions

- Testing is used to ensure a chip's quality.
- Testing is a complex and expensive task and should be dealt with at early (design) stage.
- Test strategies should be evaluated with a solid and overall economics model.