TESTING AND TESTABLE DESIGN OF DIGITAL SYSTES

Kewal K. Saluja
University of Wisconsin - Madison

Motivation, Fault Models and some Callenges

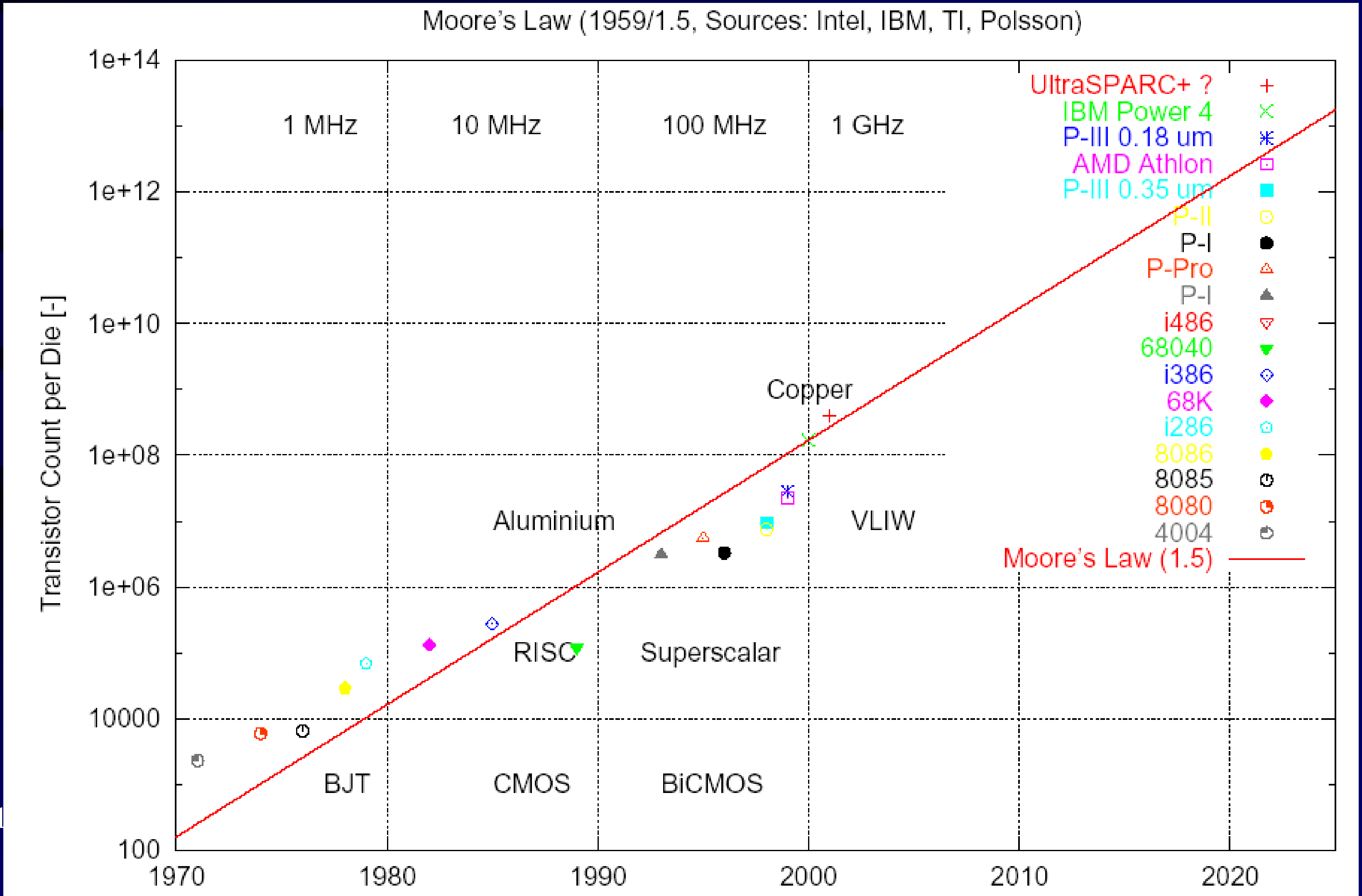
Overview

- Motivation
  - Technology, Test cost, and VLSI realization process
  - Contract between design house and fab vendor
  - Need for testing and Levels of testing – rule of 10 (or 20)
  - Cost and role of testing
- Fault Modeling
  - Why model faults and some real defects in VLSI and PCB
  - Common fault models
  - Net list description, stuck-at faults and number of faults
  - Method to reduce fault list
  - Other faults – transistor faults
- Some Challenges in testing
  - Test application time and Test data compaction
  - Test scheduling

Motivation: Moore’s Law

Complexity Growth of VLSI circuits

Source (Copp, *Int. AOC EW Conf., 2002*)



Thermal Effect of Moore's law

- Moore's law

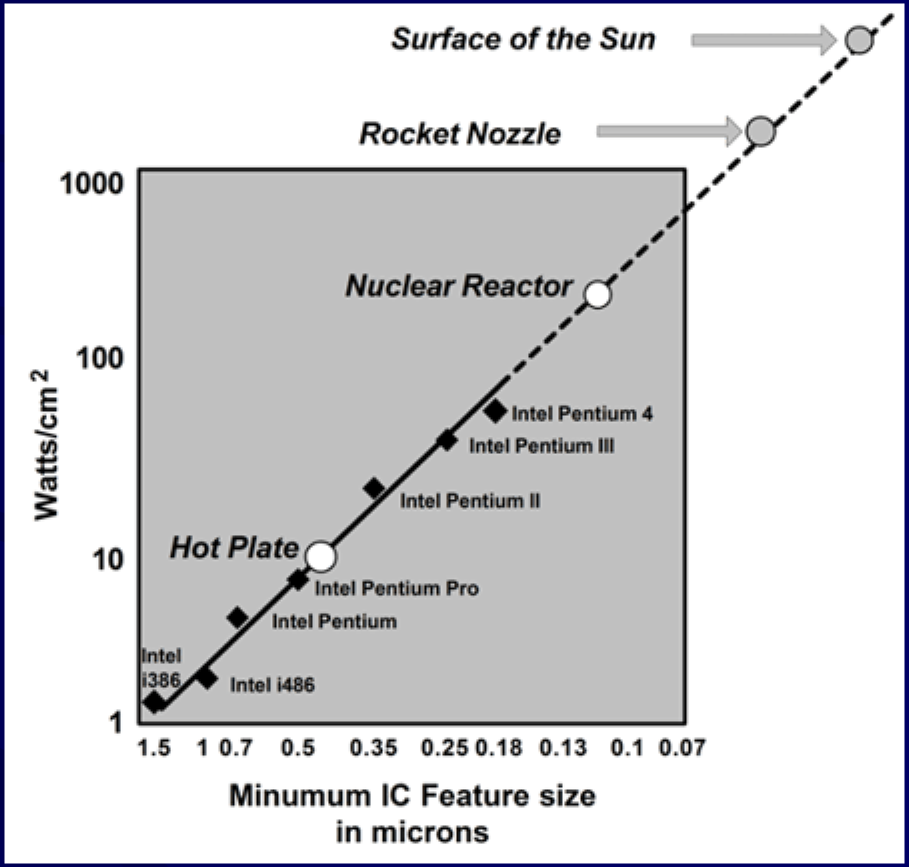
  Greater packaging densities

  ↓

  Higher power densities

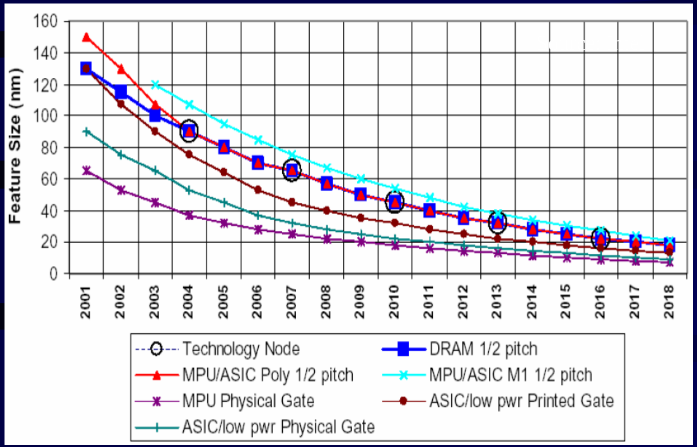
  ↓

  Higher temperature

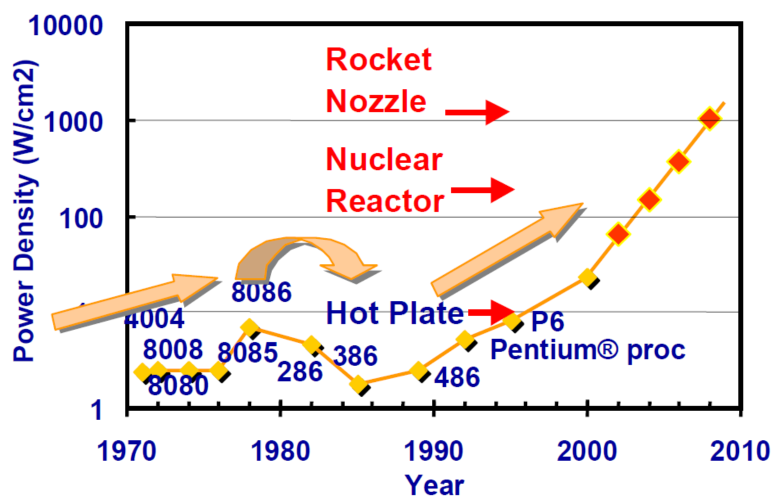


- Effects
  - Reliability
  - Performance
  - Power
  - Cooling cost

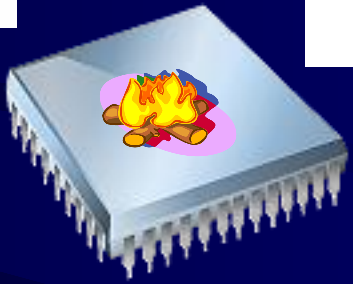
Introduction: Challenges under deep submicron technologies (Yao)

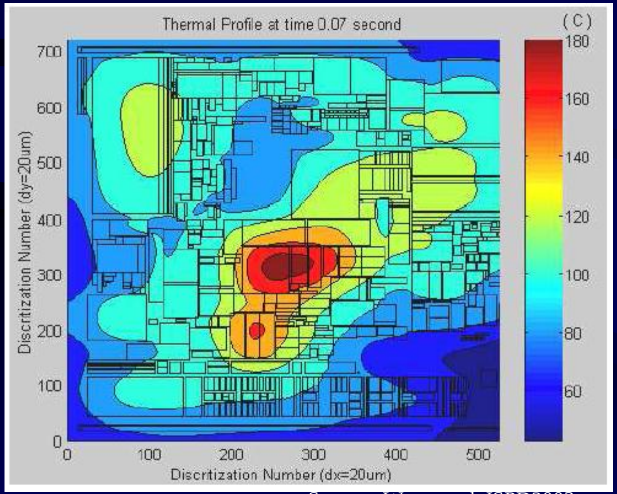


Chip size decreases



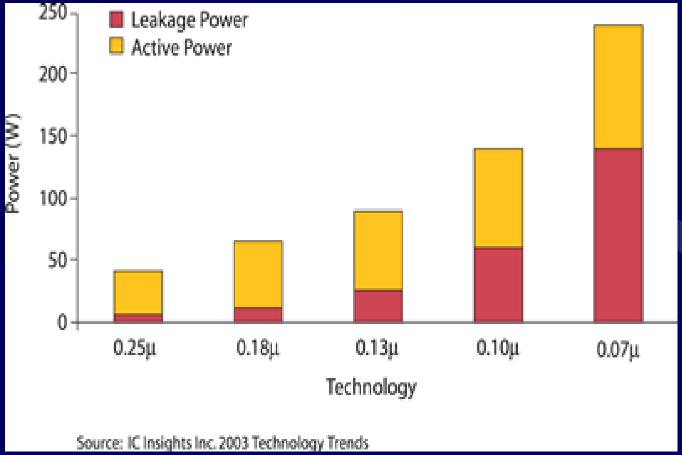
Power density increases




Thermal Profile at time 0.07 second
(C)
180
160
140
120
100
80
60
Discritization Number (dy=20um)
700
600
500
400
300
200
100
0
0
100
200
300
400
500
Discritization Number (dx=20um)

Source: Wang et al. ISPD2003

Chip becomes hotter



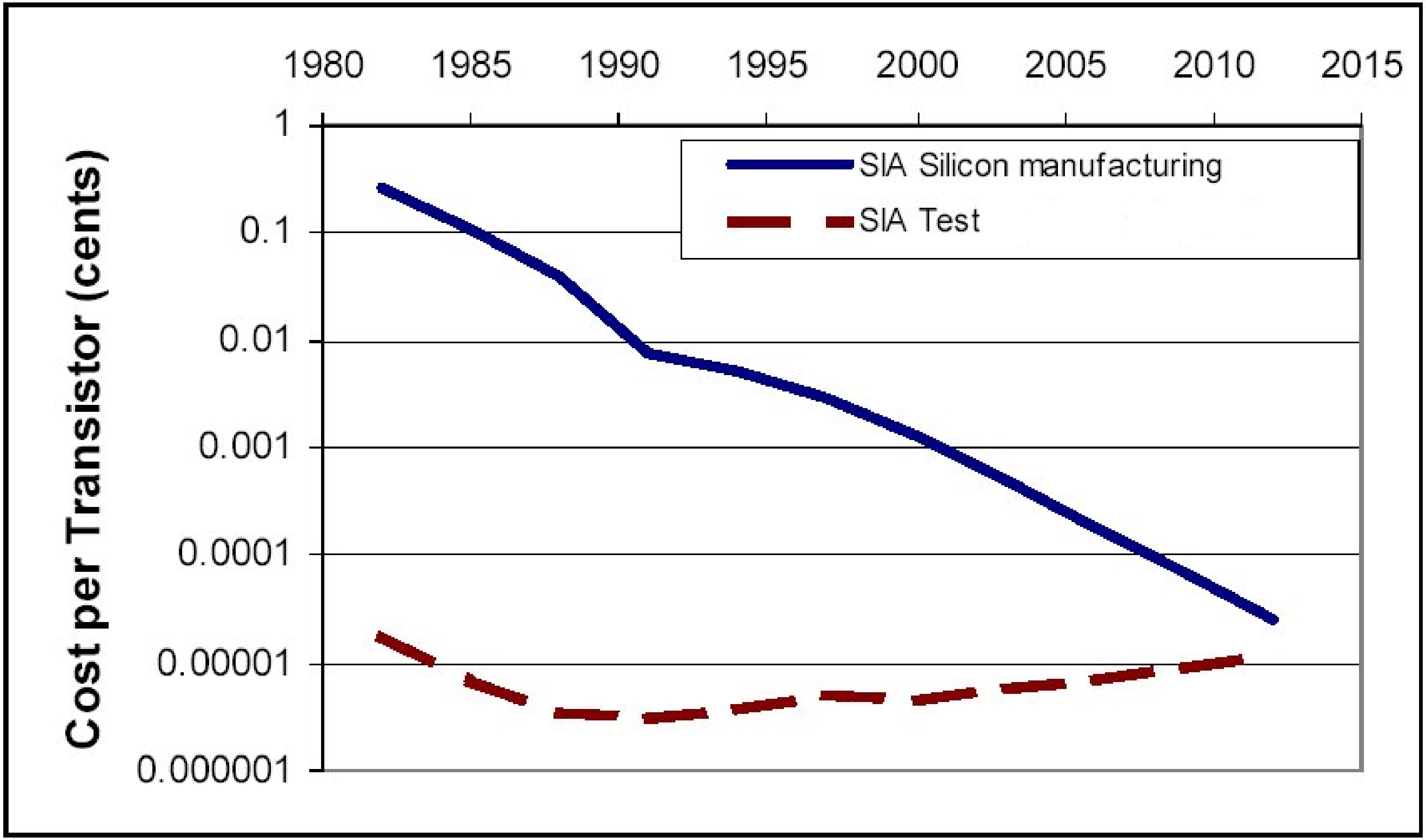
Source: IC Insights Inc. 2003 Technology Trends

Leakage power make it worse

Microprocessor Cost per Transistor

Cost of testing will *EXCEED* cost of design/manufacturing

(Source: ITR-Semiconductor, 2002)



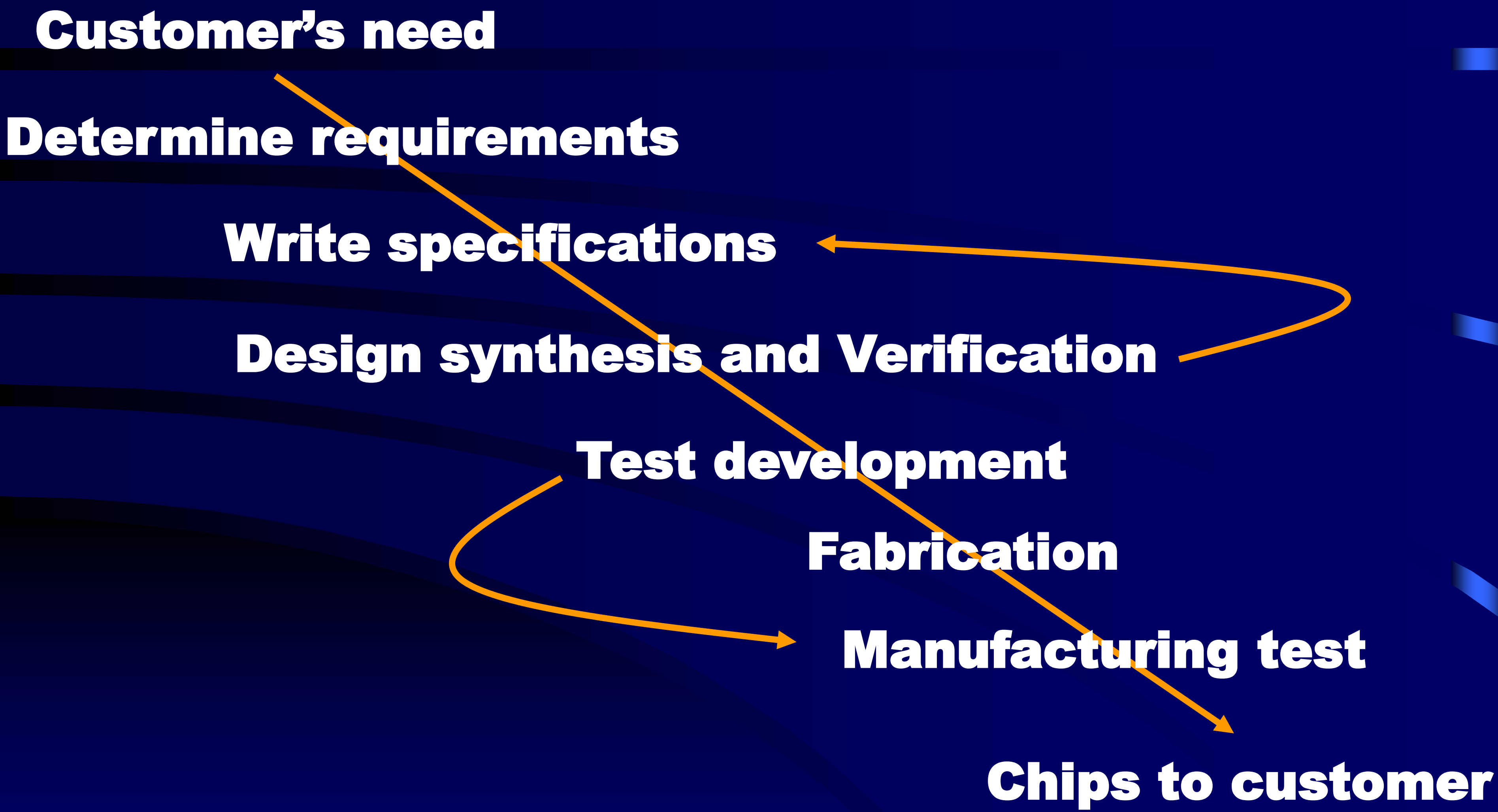
Present and Future

Technology Directions: SIA Roadmap

Year	1999	2002	2005	2008	2011	2014
Feature size (nm)	180	130	90	45	32	22
Mtrans/cm^2	7	14-26	47	115	284	701
Chip size (mm^2)	170	170-214	235	269	308	354
Signal pins/chip	768	1024	1024	1280	1408	1472
Clock rate (MHz)	600	800	1100	1400	1800	2200
Wiring levels	6-7	7-8	8-9	9	9-10	10
Power supply (V)	1.8	1.5	1.2	0.9	0.6	0.6
High-perf power (W)	90	130	160	170	174	183
Battery power (W)	1.4	2.0	2.4	2.0	2.2	2.4

http://www.itrs.net/ntrs/publntrs.nsf

VLSI Realization Process



Customer's need

Determine requirements

Write specifications

Design synthesis and Verification

Test development

Fabrication

Manufacturing test

Chips to customer

Contract between a design house and a fab vendor

- Design is complete and checked (verified)
- Fab vendor: How will you test it?
- Design house: I have checked it and …
- Fab vendor: But, how would you test it?
- Desing house: Why is that important? It is between I and my clients – it is none of your business
- Fab vendor – Sorry you can take your business some where else.

complete the story and determine the reasons for the importance of test generation etc.

Contract between design …

Hence:

- “Test” must be comprehensive
- It must not be “too long”

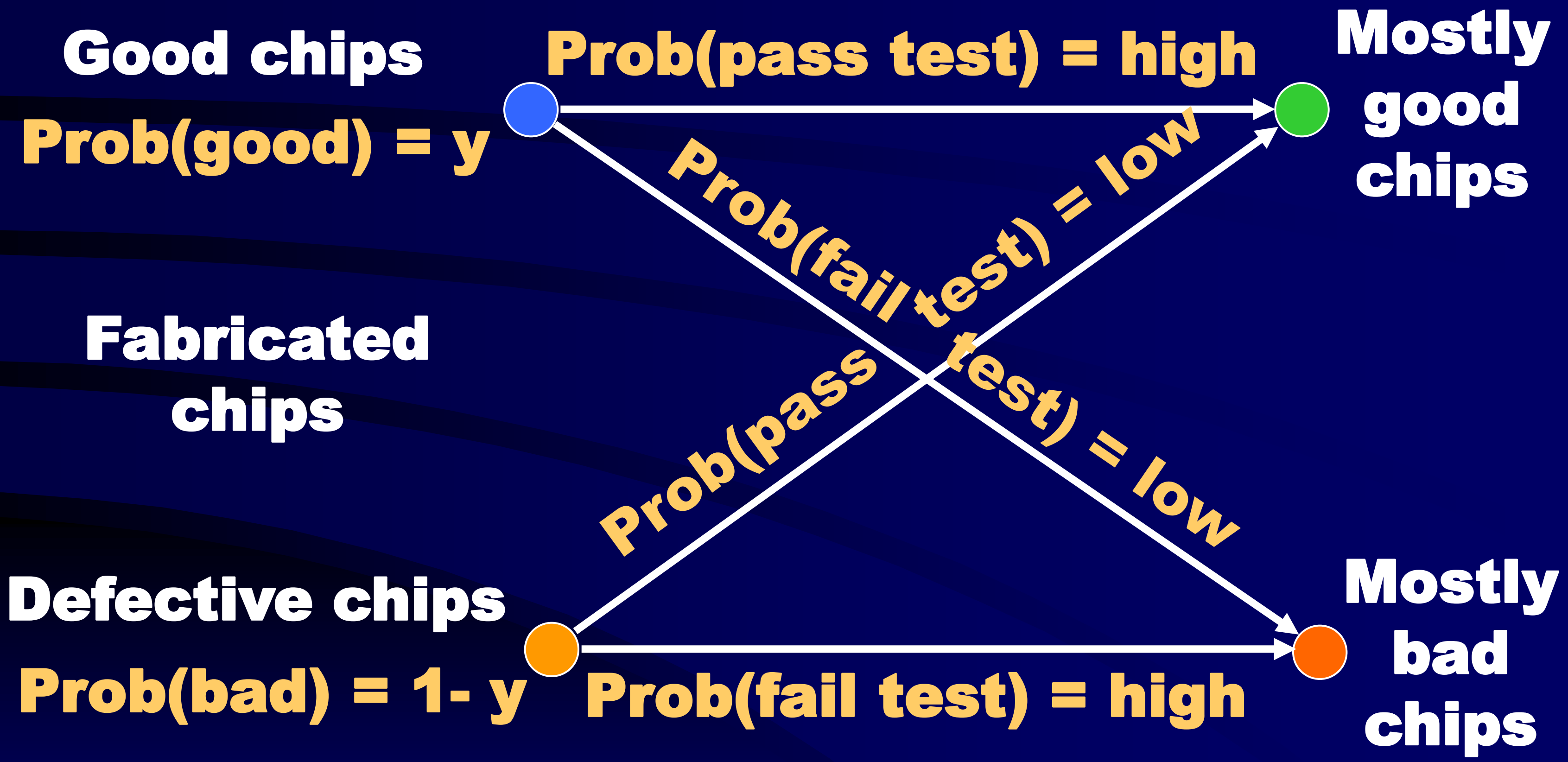
Issues:

- Model possible defects in the process
  - Understand the process
- Develop logic simulator and fault simulator
- Develop test generator
- Methods to quantify the test efficiency

Need for testing

- Functionality issue
  - Does the circuit (large or small) work?
- Density issue
  - Higher density ⇒ higher failure probability
- Application issue
  - Life critical applications
- Maintenance issue
  - Need to identify failed components
- Cost of doing business
- What does testing achieve?
  - Discard only the “bad product”? – see next three slides

Testing as Filter Process



Good chips

Prob(good) = y

Prob(pass test) = high

Mostly good chips

Prob(fail test) = low

Fabricated chips

Prob(pass test) = low

Defective chips

Prob(bad) = 1- y

Prob(fail test) = high

Mostly bad chips

Levels of testing (1)

- Levels
  - Chip
  - Board
  - System
    - Boards put together
    - System-on-Chip (SoC)
  - System in field
- Cost – Rule of 10
  - It costs 10 times more to test a device as we move to higher level in the product manufacturing process

Levels of testing (2)

- Other ways to define levels – these are important to develop correct “fault models” and “simulation models”
  - Transistor
  - Gate
  - RTL
  - Functional
  - Behavioral
  - Architecture
- Focus: Chip level testing – gate level design

Cost of Testing

- *Design for testability* (DFT)
  - Chip area overhead and yield reduction
  - Performance overhead
- Software processes of test
  - Test generation and fault simulation
  - Test programming and debugging
- Manufacturing test
  - *Automatic test equipment* (ATE) capital cost
  - Test center operational cost

Roles of Testing

- Detection: Determination whether or not the *device under test* (DUT) has some fault.
- Diagnosis: Identification of a specific fault that is present on DUT.
- Device characterization: Determination and correction of errors in design and/or test procedure.
- *Failure mode analysis* (FMA): Determination of manufacturing process errors that may have caused defects on the DUT.

Why Model Faults?

- I/O function tests inadequate for manufacturing (functionality versus component and interconnect testing)
- Real defects (often mechanical) too numerous and often not analyzable
- A fault model identifies targets for testing
- A fault model makes analysis possible
- Effectiveness measurable by experiments

Some Real Defects in Chips

- **Processing defects**
  - **Missing contact windows**
  - **Parasitic transistors**
  - **Oxide breakdown**
  - **. . .**
- **Material defects**
  - **Bulk defects (cracks, crystal imperfections)**
  - **Surface impurities (ion migration)**
  - **. . .**
- **Time-dependent failures**
  - **Dielectric breakdown**
  - **Electromigration**
  - **NBTI (negative bias temperature instability)**
  - **. . .**
- **Packaging failures**
  - **Contact degradation**
  - **Seal leaks**
  - **. . .**

Ref.: M. J. Howes and D. V. Morgan, *Reliability and Degradation - Semiconductor Devices and Circuits,* Wiley, 1981.

+ more recent defect types

Netlist Format: Two Examples

UW format

```
#  gate  connected to
1   PI     4, 5 ;
2   PI     3, 6 ;
3  not     5 ;
4  not      6 ;
5  and     7 ;
6  and      7 ;
7   or   8 ;
8   PO ;
```

ISCAS format

```
output = gate(inputs)
 INPUT(G1)
 INPUT(G2)
OUTPUT(G7)
G3 = NOT(G2)
G4 = NOT(G1)
G5 = AND(G1, G3)
G6 = AND(G2, G4)
G7 = OR(G5, G6)
```

Common Fault Models

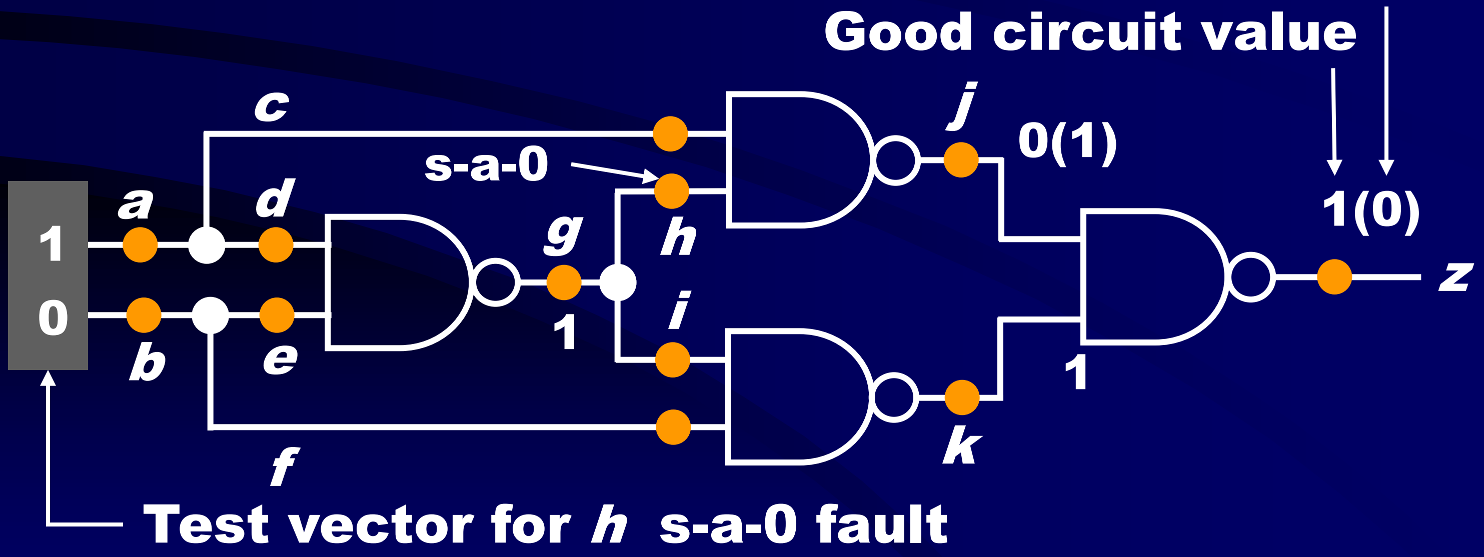
- Single stuck-at faults
- Transistor open and short faults
- Memory faults
- PLA faults (stuck-at, cross-point, bridging)
- Functional faults (processors)
- Delay faults (transition, path)
- Analog faults
- For more examples, see Section 4.4 (p. 60-70) of the book.

Stuck-at Faults

- Single stuck-at faults
- What does it achieve in practice?
- Fault equivalence
- Fault dominance and checkpoint theorem
- Classes of stuck-at faults and multiple faults

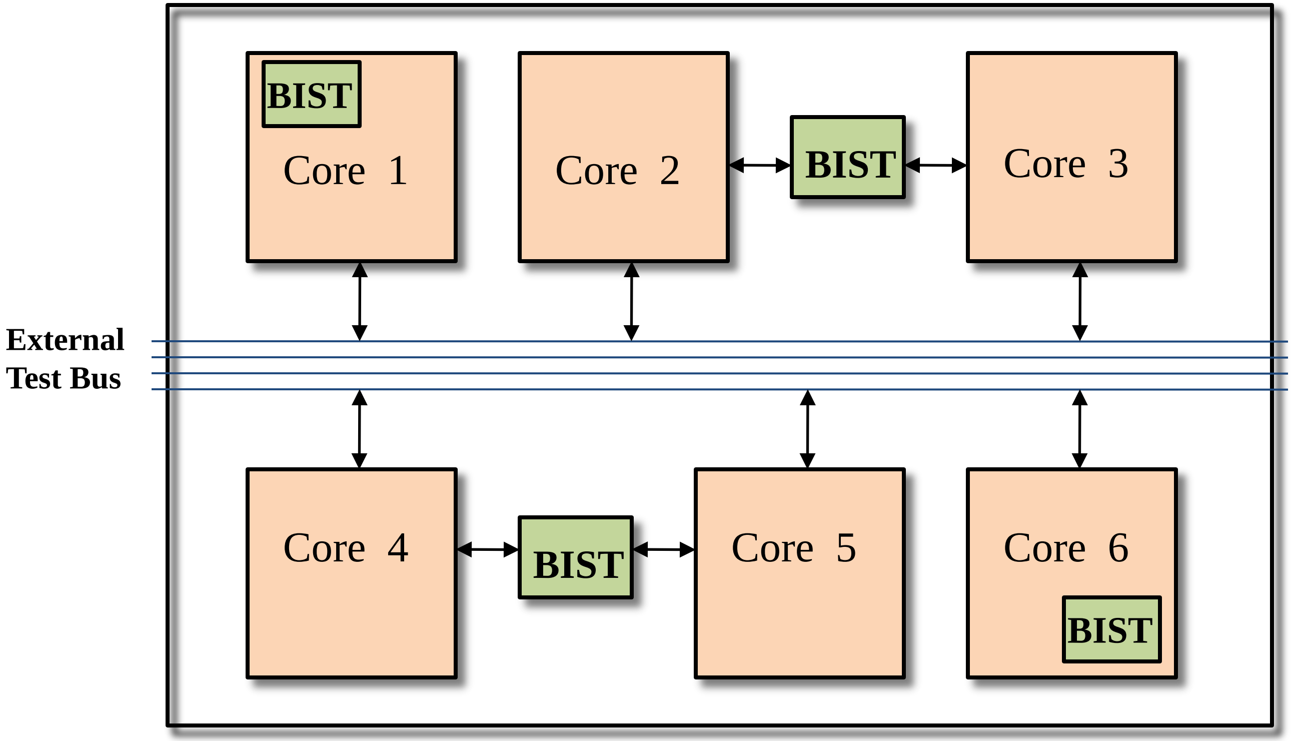
Single Stuck-at Fault

- Three properties define a single stuck-at fault
  - Only one line is faulty
  - The faulty line is permanently set to 0 or 1
  - The fault can be at an input or output of a gate
- Example: XOR circuit has 12 fault sites (●) and 24 single stuck-at faults



Test vector for *h* s-a-0 fault

Introduction: Test Scheduling



Test Scheduling: In order to reduce the test cost, the testing time must be minimized by carefully scheduling tests for cores at the system level and the tests must be scheduled without violating any constraint.

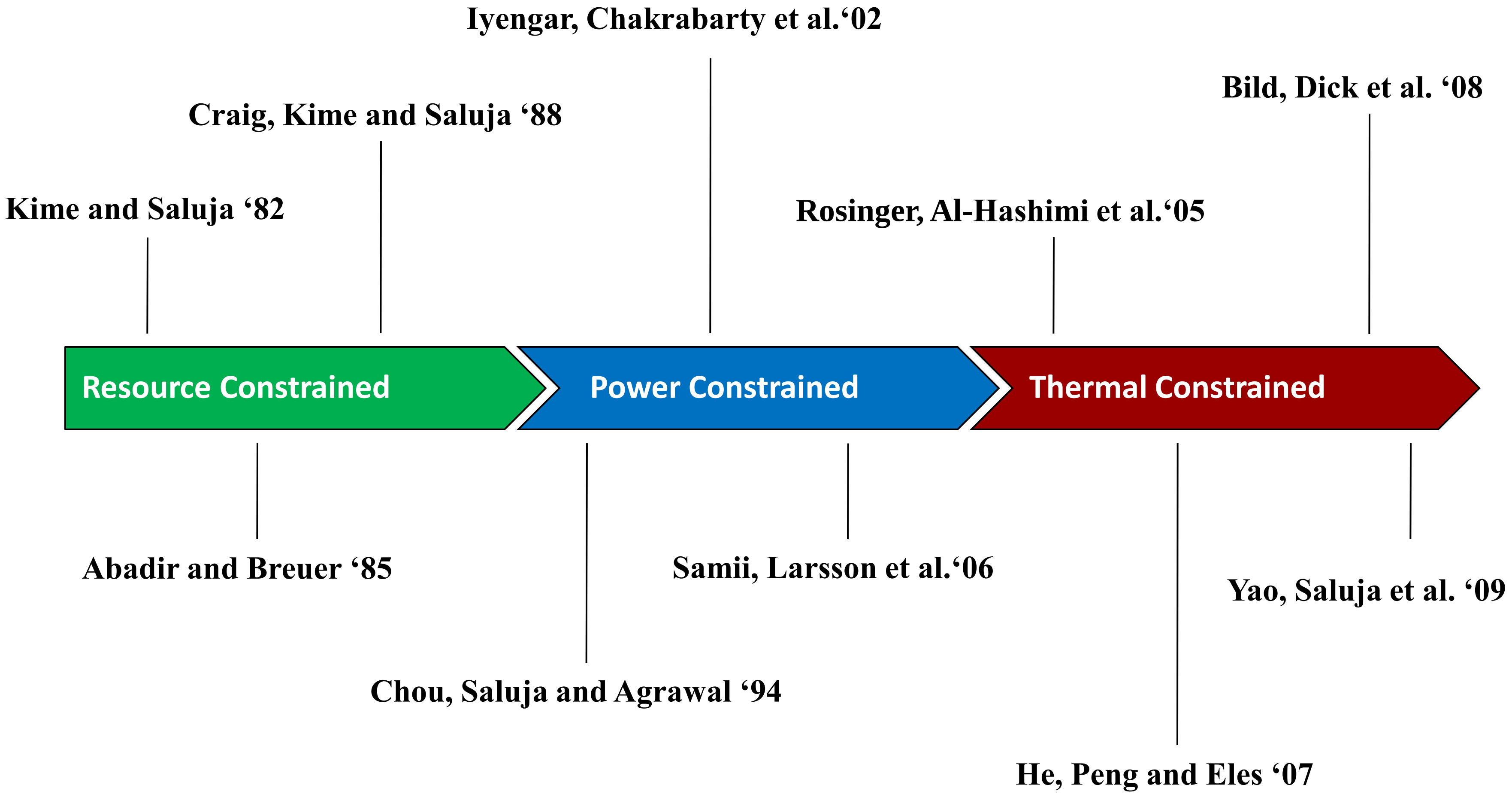
Thermal-aware test Scheduling: Problem formulation

A test set *S*, consisting of n tests t_i (i = 1 to n), needs to be applied to a device under test. Each test t_i has 1) test length L_i, 2) test power consumption P_i, 3) test compatibility with a set of known tests. (A test t_i is compatible with test t_j if there is no resource conflict between them and they can be applied to the device simultaneously.)

The test scheduling problem is to find a test schedule such that:

- 1. The total test application time is minimized. Total test application time is defined as the earliest time at which all tests have completed.
- 2. There is no resource conflict between any two simultaneously running tests.
- 3. The total power consumption of the device at all times is smaller than the power constraint, P_{max}
- 4. The temperature at every location in the device at all times is smaller than the temperature constraint, T_{max}

Introduction: History

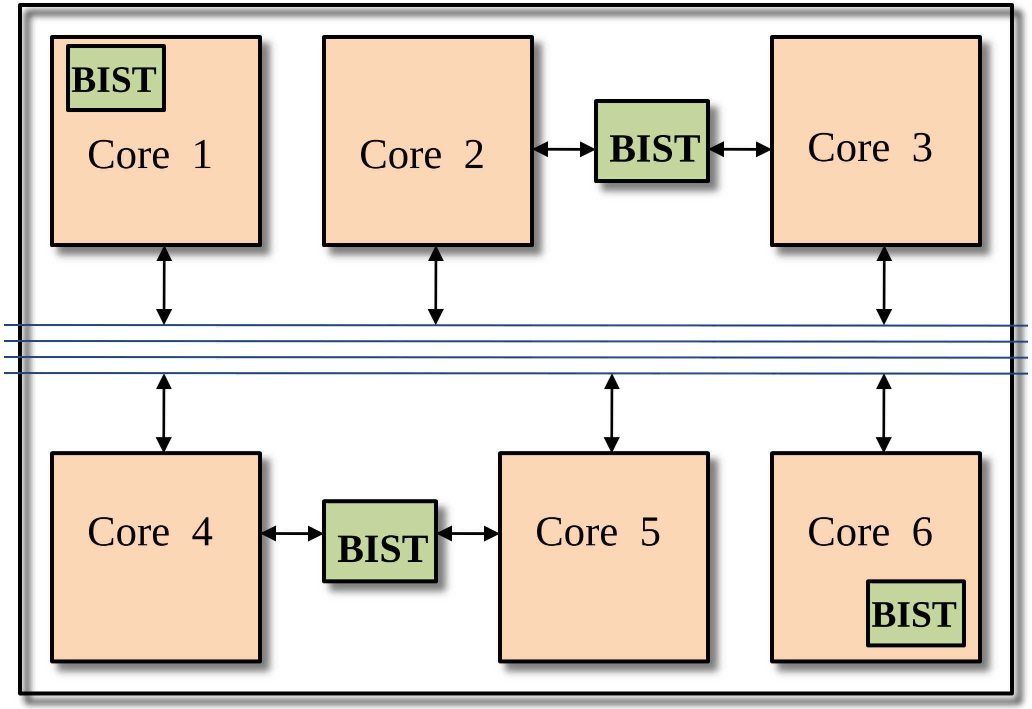


Resource constrained Test scheduling

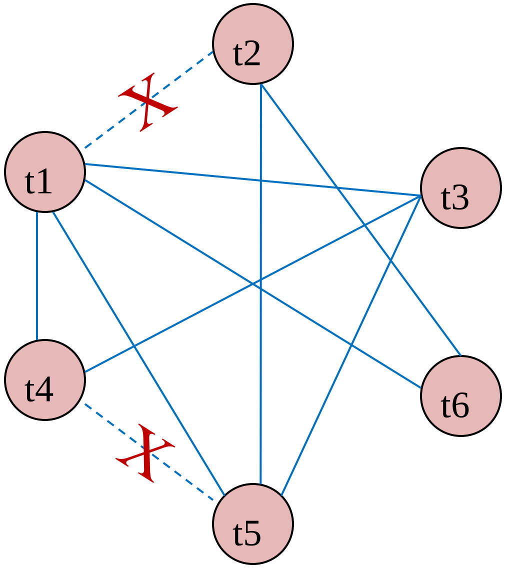
Goal: Minimize total test time under resource constraints

- Test resource conflict
  - Test pattern generator
  - Test response compactor
  - Shared test path
- Solutions:
  - Graph algorithms
    - Clique cover
    - Graph coloring
  - Combinational optimization

Resource constrained Test scheduling: Example

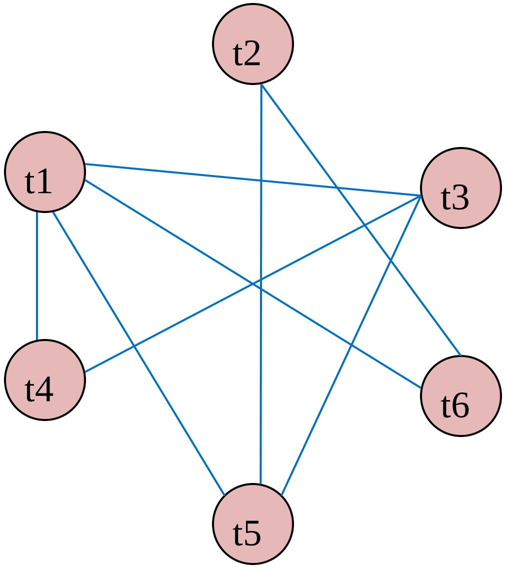


Test compatibility graph (TCG)



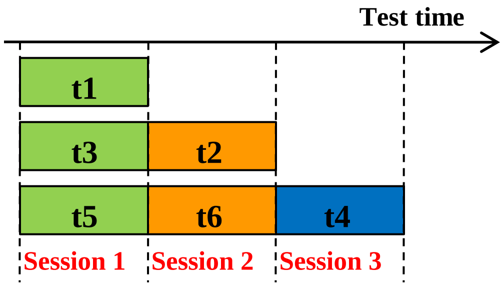
Resource constrained Test scheduling: Example

- Clique cover heuristic
  - A *clique* is a maximal complete subgraph of a graph.
  - Goal: cover all nodes by minimum number of cliques.
  - Solution:
    - (1,3,5)
    - (2,6)
    - (4)

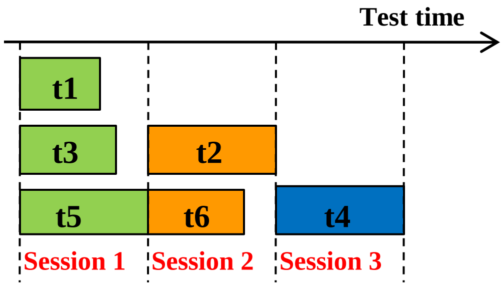


Resource constrained Test scheduling: Test session

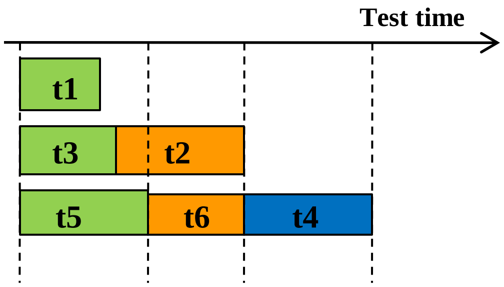
- Test Session
  - A subset of the test set such that all the tests in the test session are compatible.
  - Next test session can start only after previous test session is completed.



Test scheduling with equal length tests



Test scheduling with unequal length tests



Better solution For unequal length tests

Power constrained Test scheduling

Goal: Minimize total test time under resource and power constraints

- Algorithms:
  - Graph based
  - Rectangle/bin packing
  - Mixed Integer Linear Programming (MILP)
  - Simulated annealing
  - Genetic algorithm
  - Ant colony
- Applications:
  - System-on-Chip (SoC)
  - Network-on-Chip (NoC)
  - Multi-clock domain chips
  - Multi-supply and multi-voltage chips