Fault-Tolerant Computing

About This Presentation

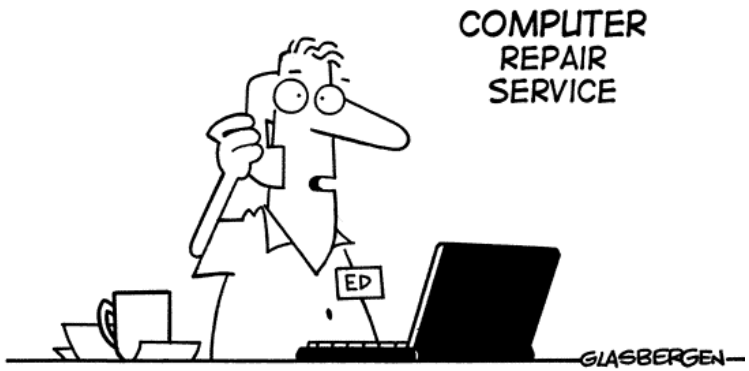
This presentation has been prepared for the graduate course ECE 257A (Fault-Tolerant Computing) by Behrooz Parhami, Professor of Electrical and Computer Engineering at University of California, Santa Barbara. The material contained herein can be used freely in classroom teaching or any other educational setting. Unauthorized uses are prohibited. © Behrooz Parhami

Edition	Released	Revised	Revised
First	Oct. 2006		

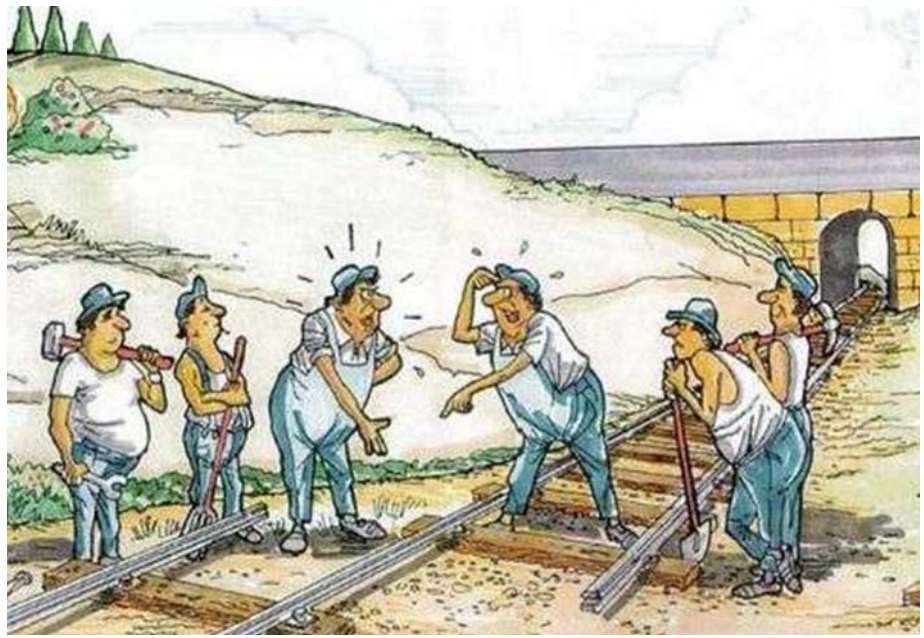
Defect Avoidance and Circumvention

"They found a defect in the new chip. Looks like someone was asleep at the itty-bitty, teeny-weeny switch."

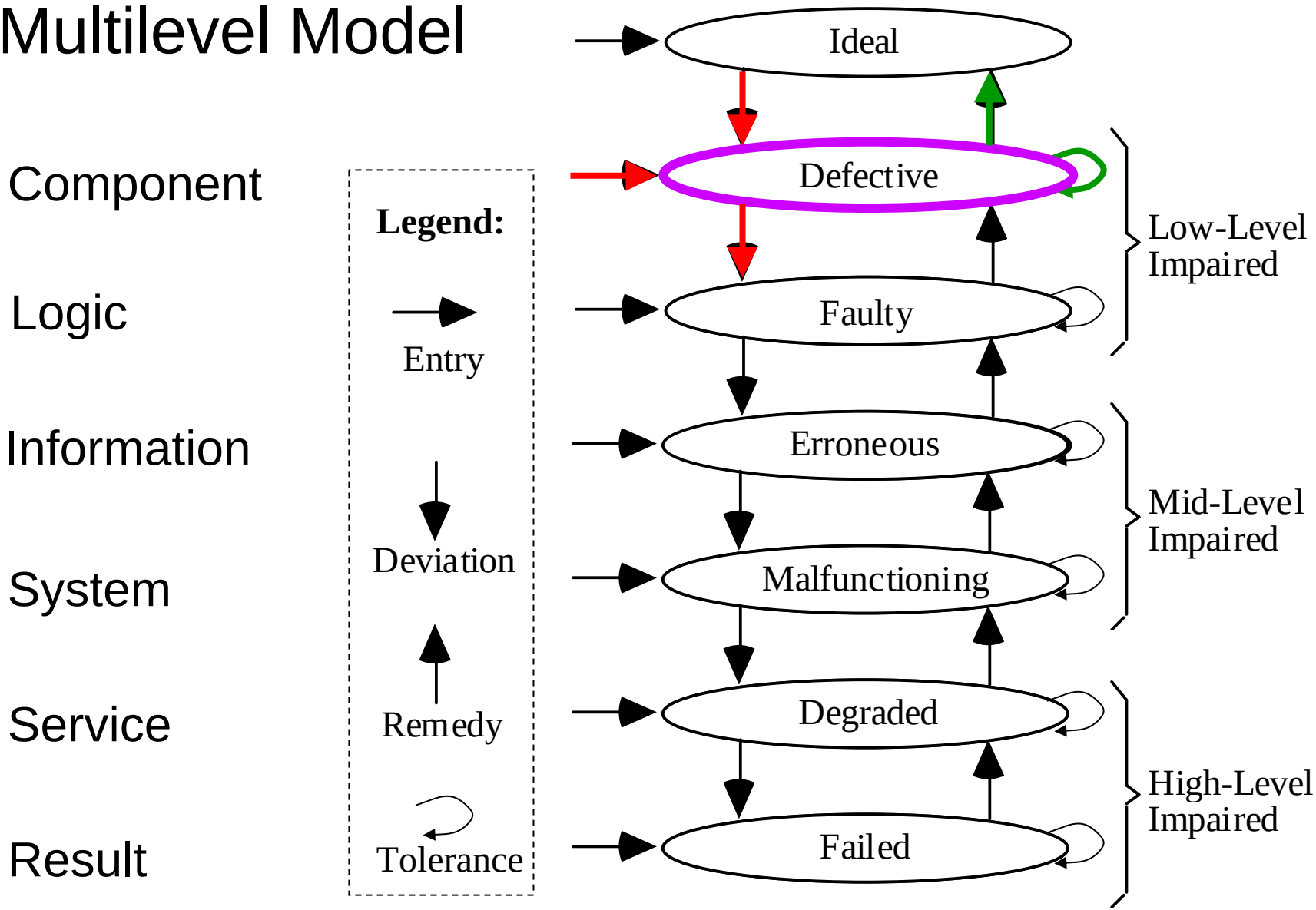
"Thinking outside of the box didn't work. Thinking inside of the box didn't work. Maybe it's a defective box!"



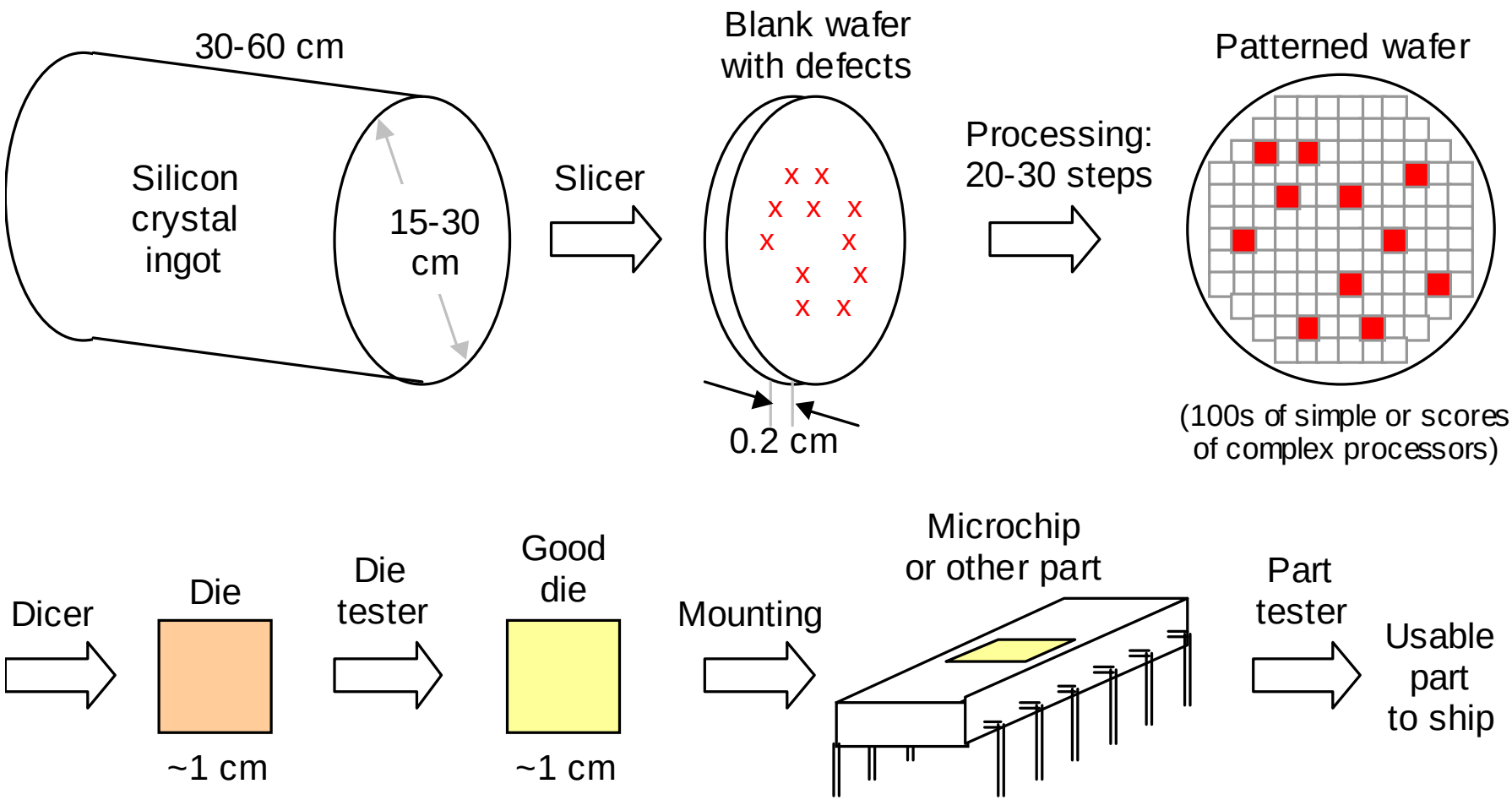
"The whole keyboard isn't damaged, just the colon. I'm referring you to a proctologist."



Multilevel Model

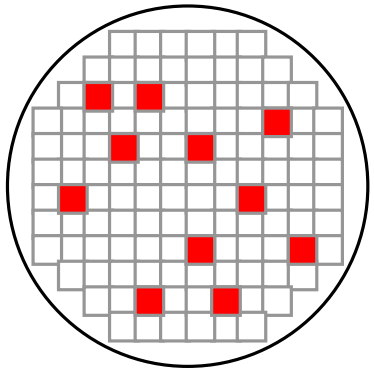


The Manufacturing Process for an IC Part

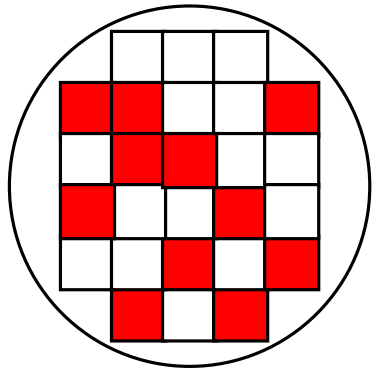


Effect of Die Size on Yield

Shown are some random defects; there are also bulk or clustered defects that affect a large region



120 dies, 109 good



26 dies, 15 good

The dramatic decrease in yield with larger dies

Die yield $=_{def}$ (Number of good dies) / (Total number of dies)

Die yield = Wafer yield $\times$ [1 + (Defect density $\times$ Die area) / a]$^{-a}$

Die cost = (Cost of wafer) / (Total number of dies $\times$ Die yield)
= (Cost of wafer) $\times$ (Die area / Wafer area) / (Die yield)

The parameter a ranges from 3 to 4 for modern CMOS processes

Effects of Yield on Testing and Part Reliability

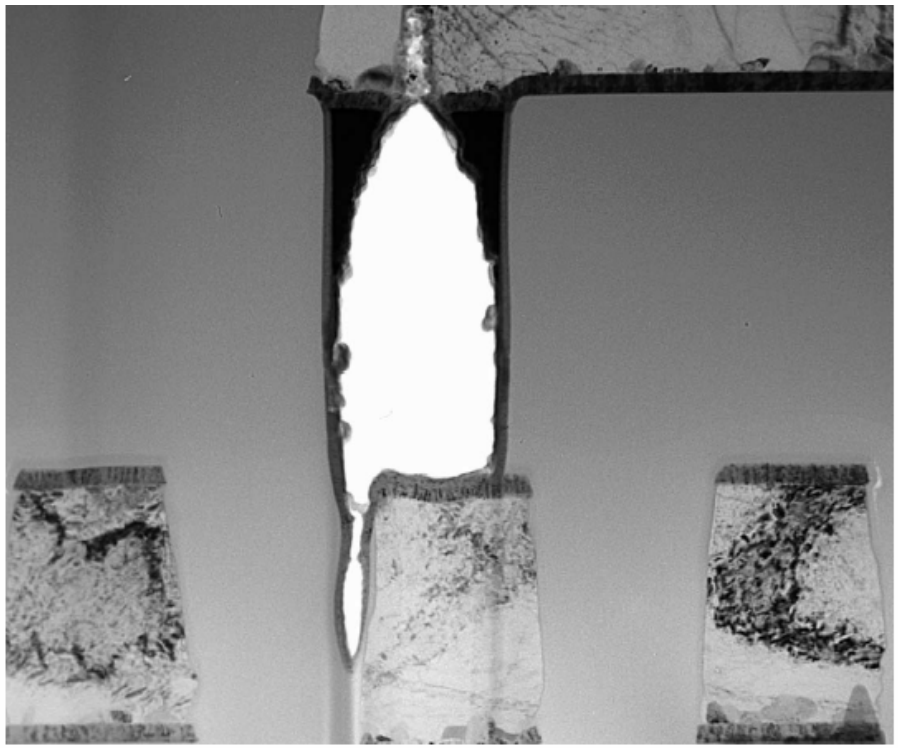
Die yield $=_{assume}$ 50%

Out of 2,000,000 dies manufactured, ≈1,000,000 are defective

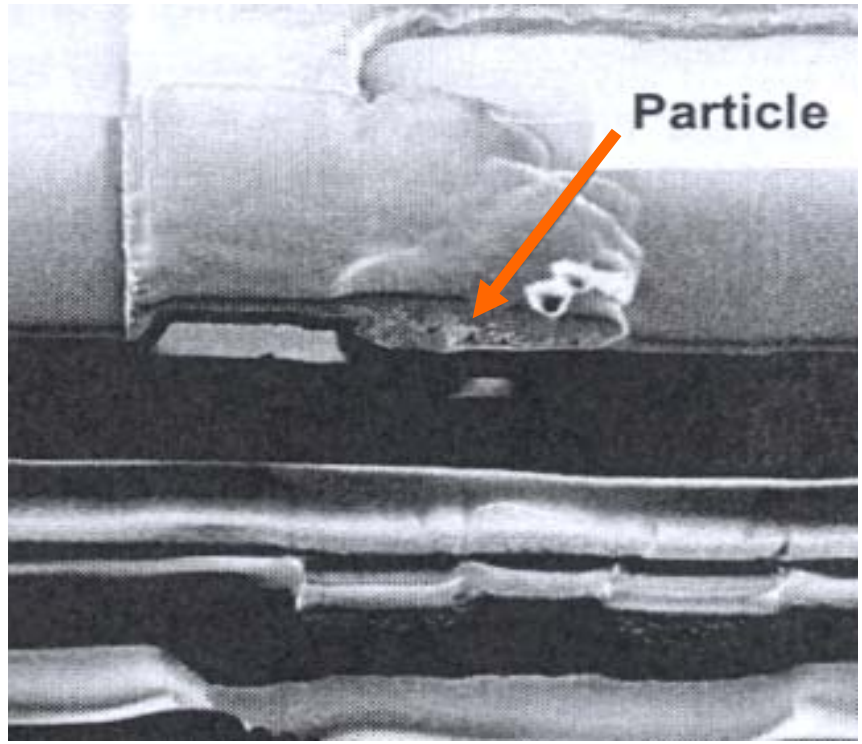
To achieve the goal of 100 defects per million (DPM) in parts shipped, we must catch 999,900 of the 1,000,000 defective parts

Therefore, we need a test coverage of 99.99%

Examples of Random Defects in ICs

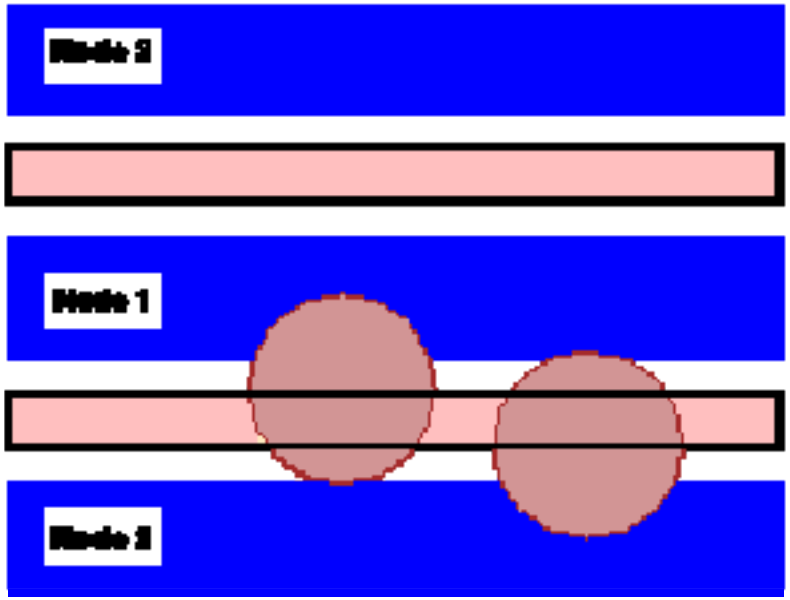


Resistive open due to unfilled via
[R. Madge et al., *IEEE D&T,* 2003]

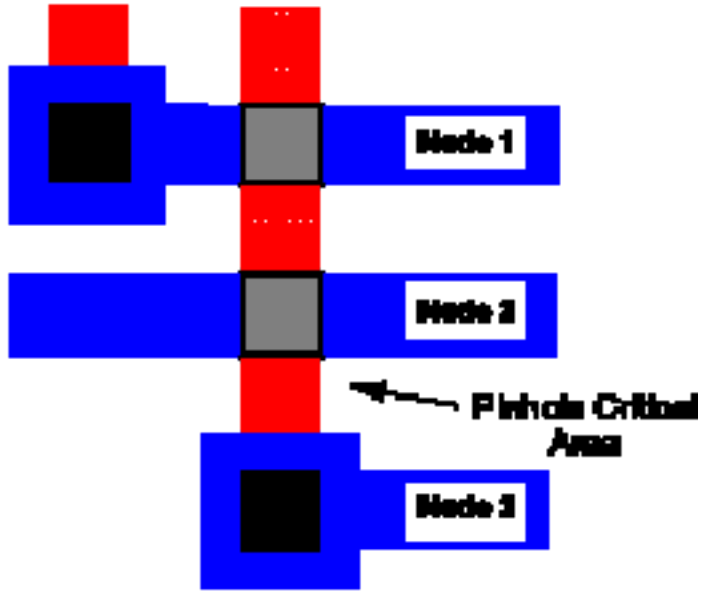


Particle embedded
between layers

Defect Modeling



(a) Extra Material Critical Area



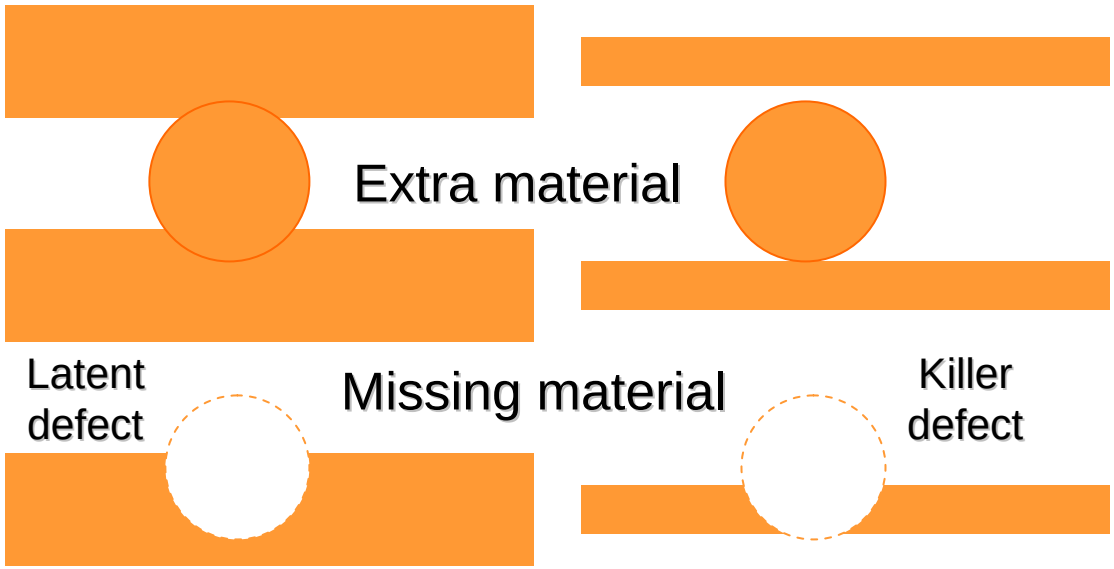
(b) Pinhole Critical Area

Extra-material defects are modeled as circular areas

Pinhole defects are tiny breaches in the dielectric

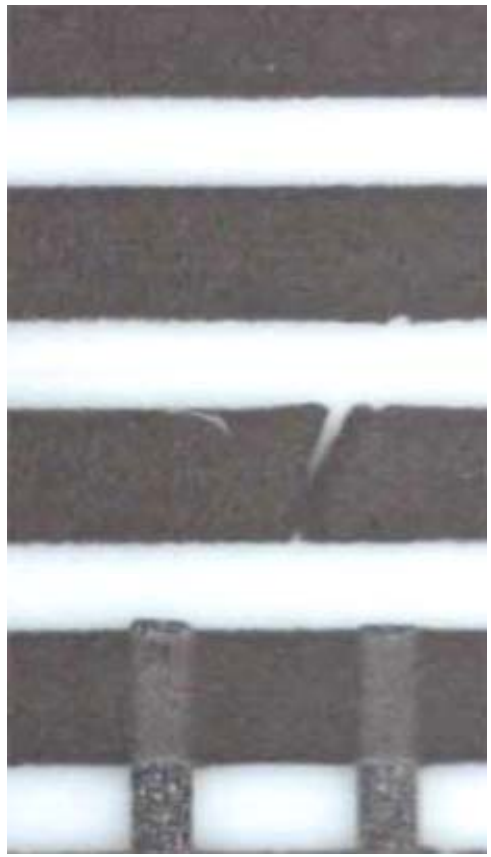
From: http://www.see.ed.ac.uk/research/IMNS/papers/IEE_SMT95_Yield/IEEAbstract.html

Sensitivity of Layouts to Defects



VLSI layout must be done with defect patterns and their impacts in mind

A balance must be struck with regard to sensitivity to different defect types

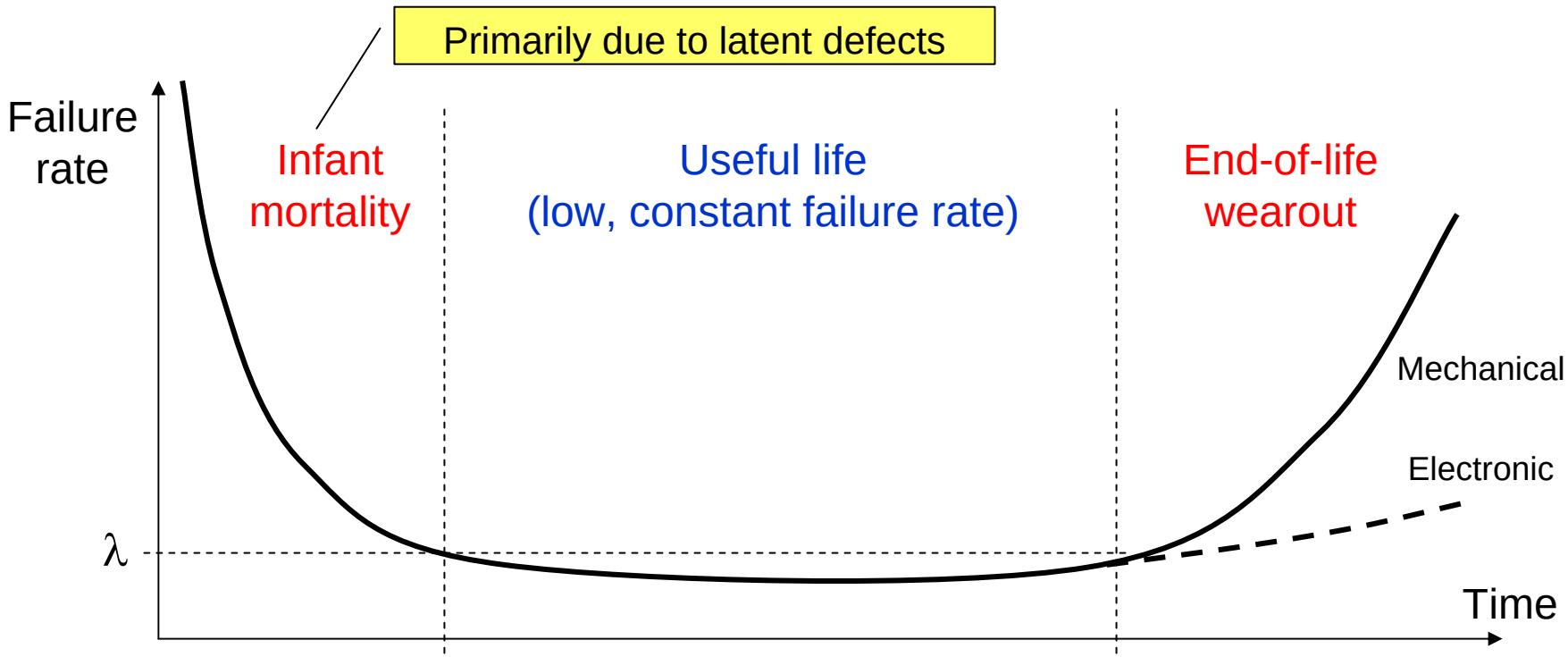


Actual photo of a missing-material defect

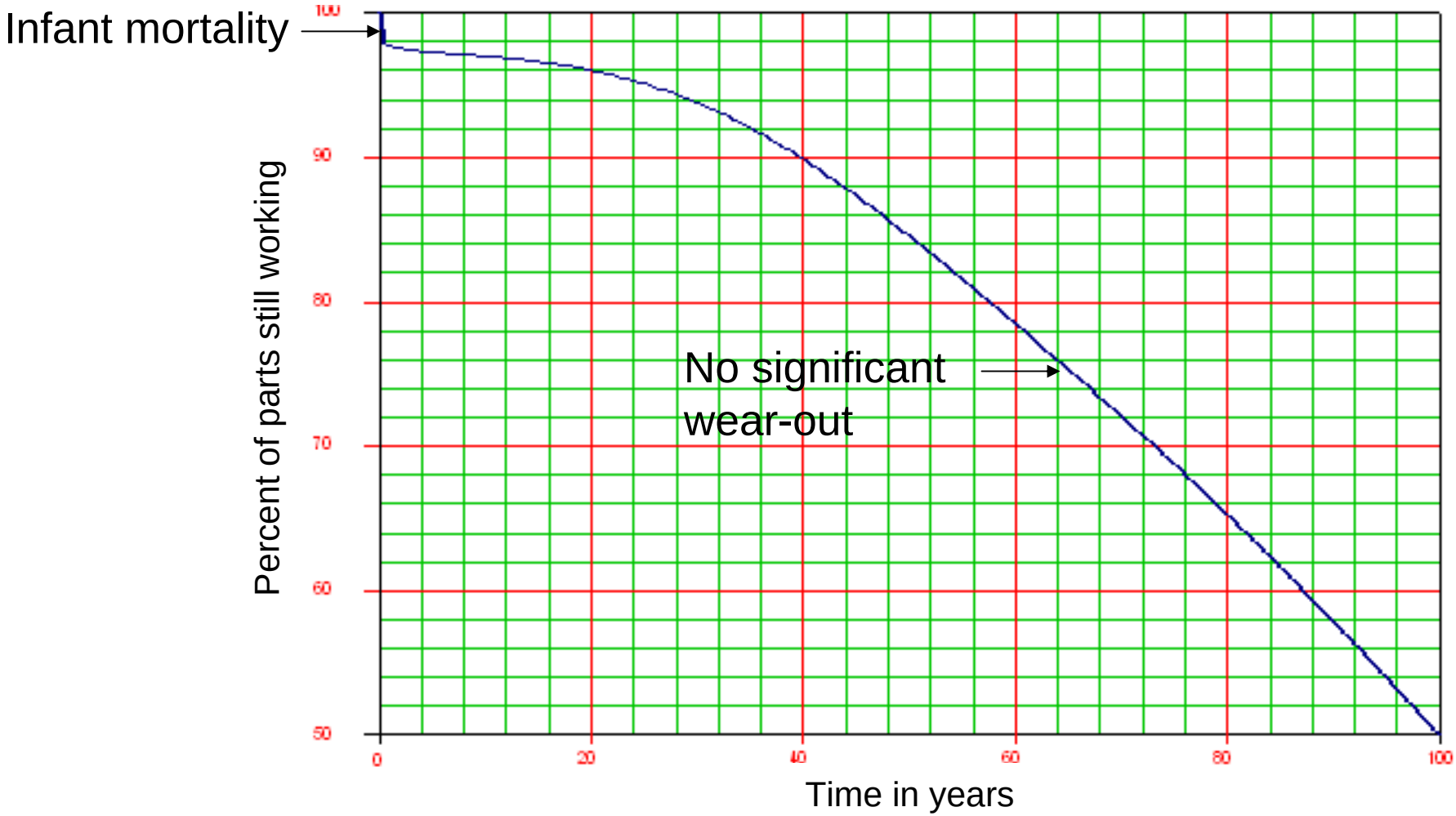
http://www.midasvision.com/v3.htm

The Bathtub Curve

Many components fail early on because of residual or latent defects
Components may also wear out due to aging (less so for electronics)
In between the two high-mortality regions lies the useful life period



Survival Probability of Electronic Components



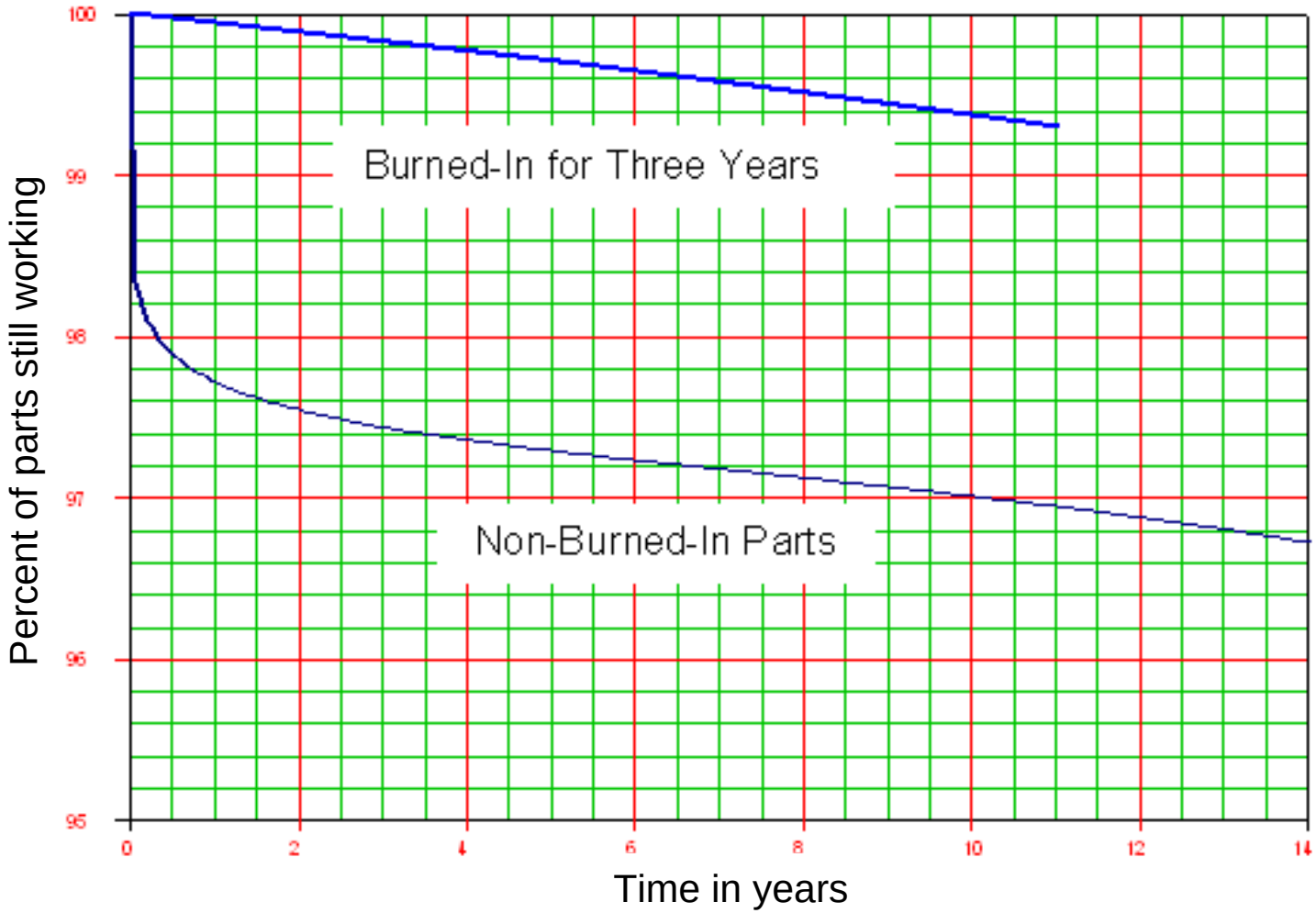
From: http://www.weibull.com/hotwire/issue21/hottopics21.htm

Burn-In and Stress Testing

Burn-in and stress tests are done in accelerated form

Difficult to perform on complex and delicate ICs without damaging good parts

Expensive “ovens” are required



From: http://www.weibull.com/hotwire/issue21/hottopics21.htm

Defect Avoidance vs. Circumvention

Defect Avoidance

Defect awareness in design, particularly layout and routing
Extensive quality control during the manufacturing process
Comprehensive screening, including burn-in and stress tests

Defect Circumvention (Removal)

Built-in dynamic redundancy on the die or wafer
Identification of defective parts (visual inspection, testing, association)
Bypassing or reconfiguration via embedded switches

Defect Circumvention (Tolerance)

Built-in static redundancy on the die or wafer
Identification of defective parts (external test or self-test)
Adjustment or tuning of redundant structures

Defect Bypassing via Reconfiguration

Works best when the system on die has regular, repetitive structure:

- Memory
- FPGA
- Multicore chip
- CMP (chip multiprocessor)

Irregular (random) logic implies greater redundancy due to replication:

- Replicated structures must not be close to each other
- They should not be very far either (wiring/switching overhead)

Defects in Memory Arrays

Defect circumvention (removal)
Provide several extra (spare) rows and/or columns
Route external connections to defect-free rows and columns

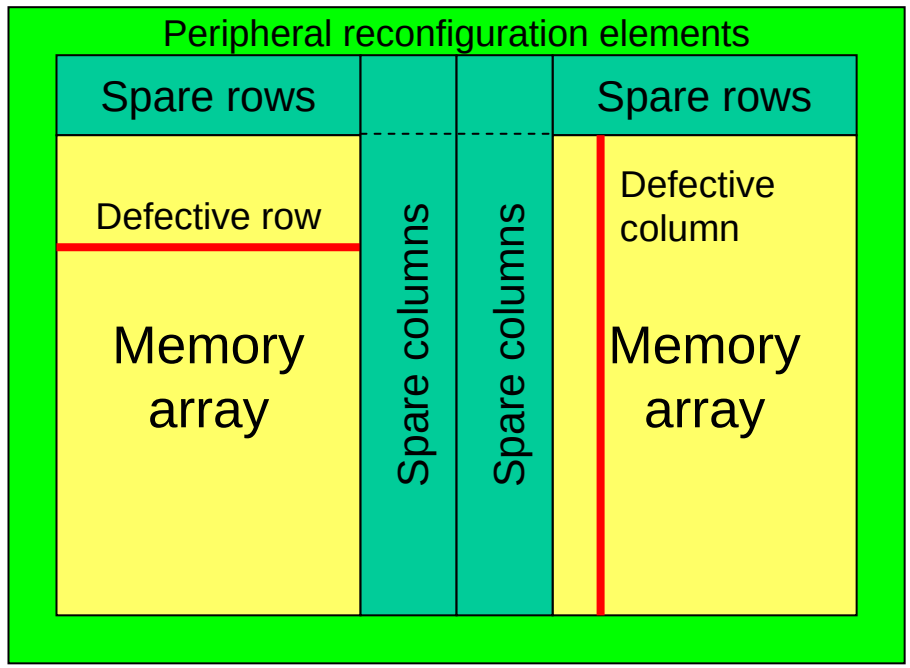
Defect circumvention (tolerance)
Error-correcting code

Methods in use since the 1970s;
e.g., IBM's defect-tolerant chip

With *m* rows and *s* spares,
can model as *m*-out-of-(*m* + *s*)

Somewhat more complex with
both spare rows and columns
(still combinational, though)

Modeling with coded scheme
to be discussed at the info level



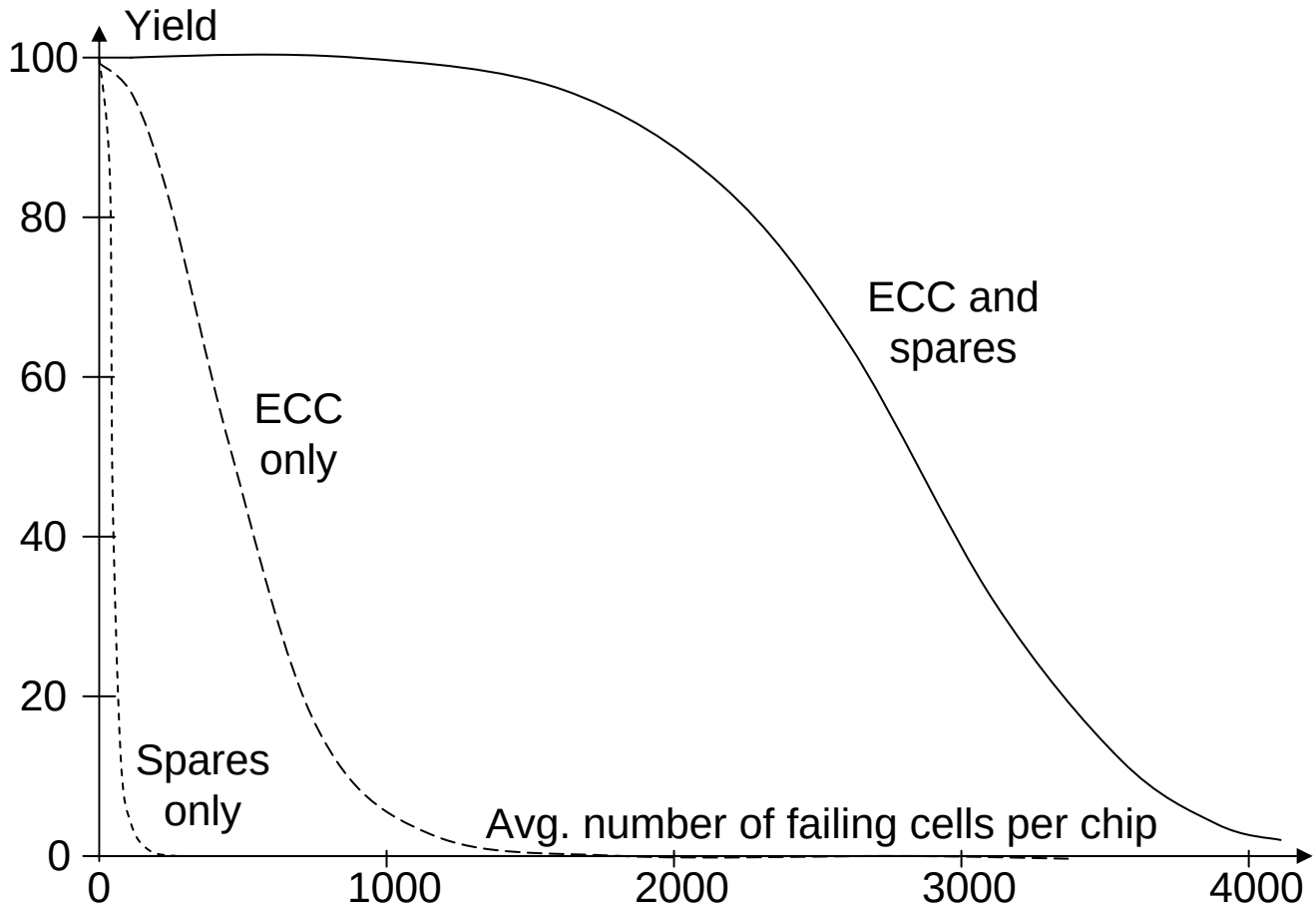
Yield Improvement in Memory Arrays

Example of IBM's experimental 16 Mb memory chip
Combines the use of spare rows/columns in memory arrays with ECC

Four quadrants, each with 16 spare rows & 24 spare columns

ECC corrects any single error via 9 check bits (137 data bits)

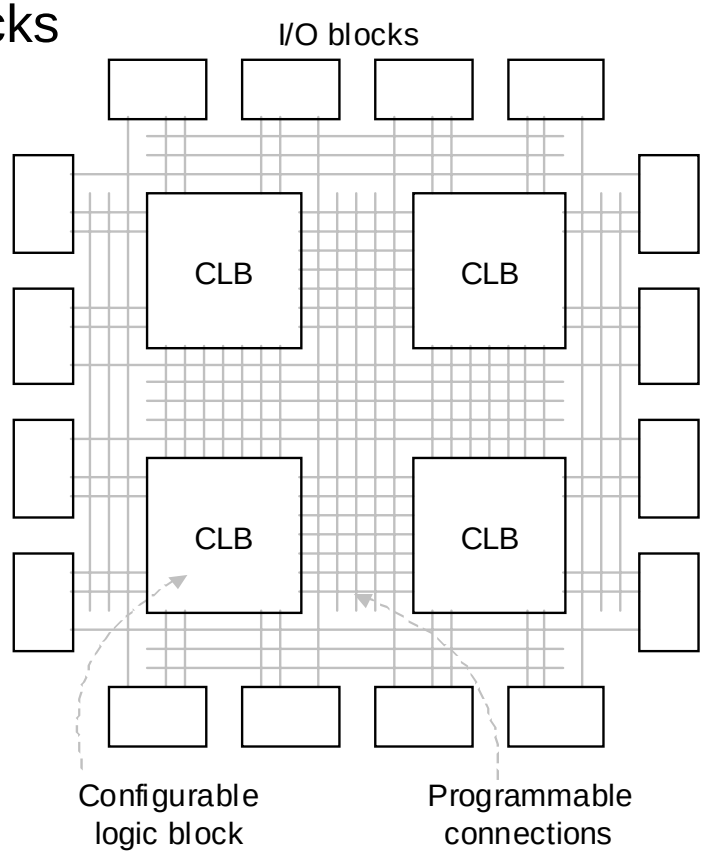
Bits assigned to the same word are separated by 8 bit positions



Defects in FPGAs

Defect circumvention (removal)
Provide several extra (spare) CLBs, I/O blocks, and connections
Route external connections to available blocks

Defect circumvention (tolerance)
Not applicable



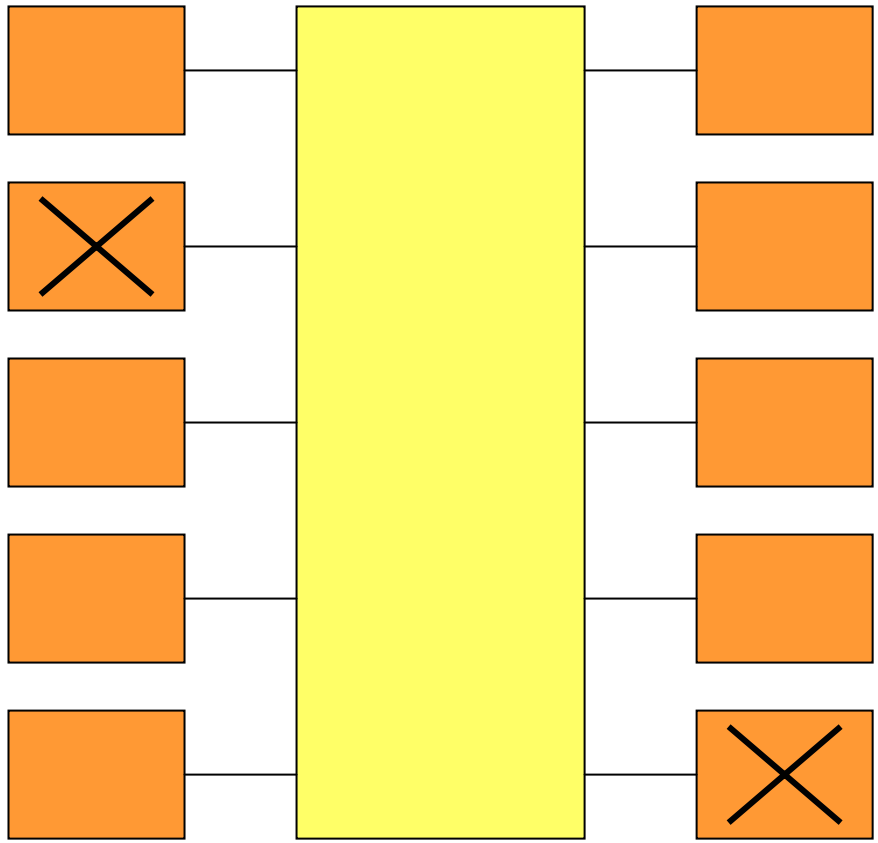
(

Defects in Multicore Chips or CMPs

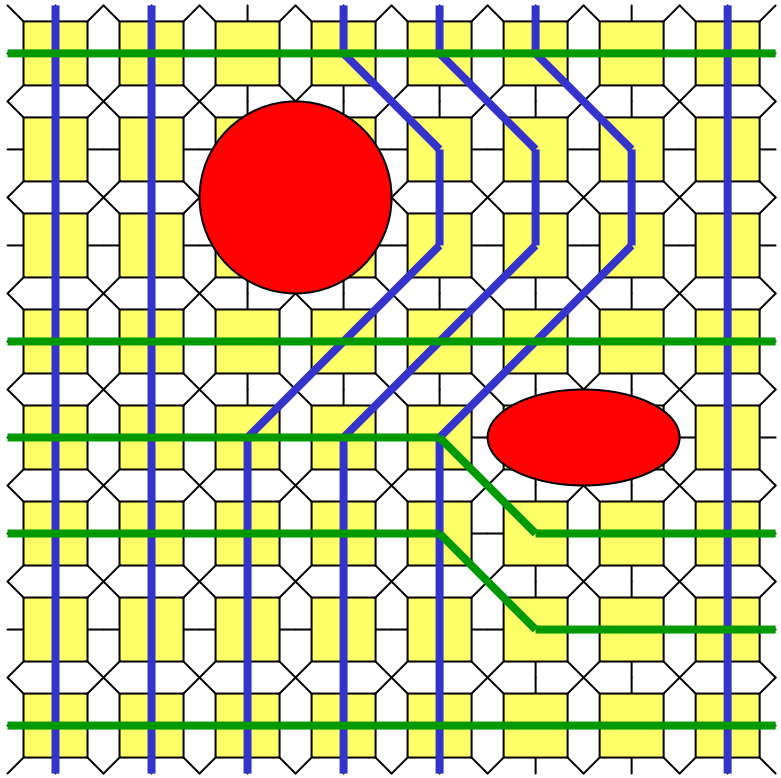
Defect circumvention (removal)
Similar to FPGAs, except that processors are the replacement entities

Interprocessor interconnection network is the main challenge

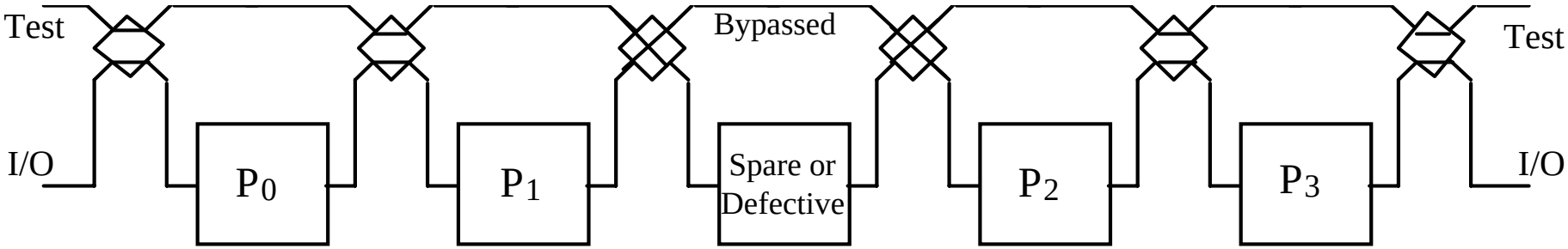
Will discuss the switching and reconfiguration aspects in more detail when we get to the malfunction level in our multilevel model



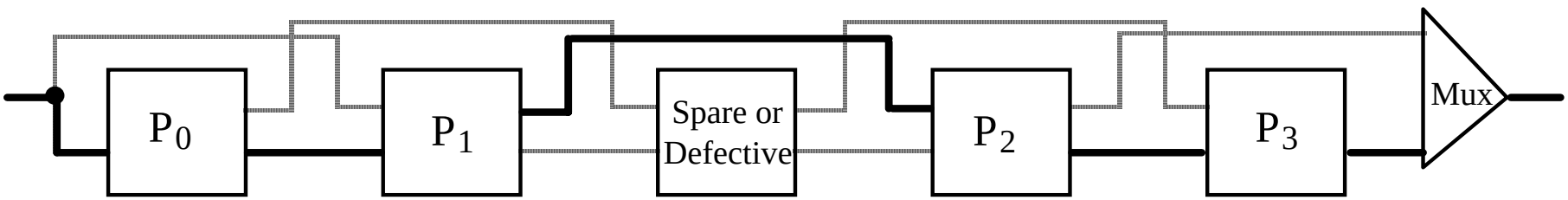
Circumventing Defects in Processor Arrays



Defect Tolerance Schemes for Linear Arrays

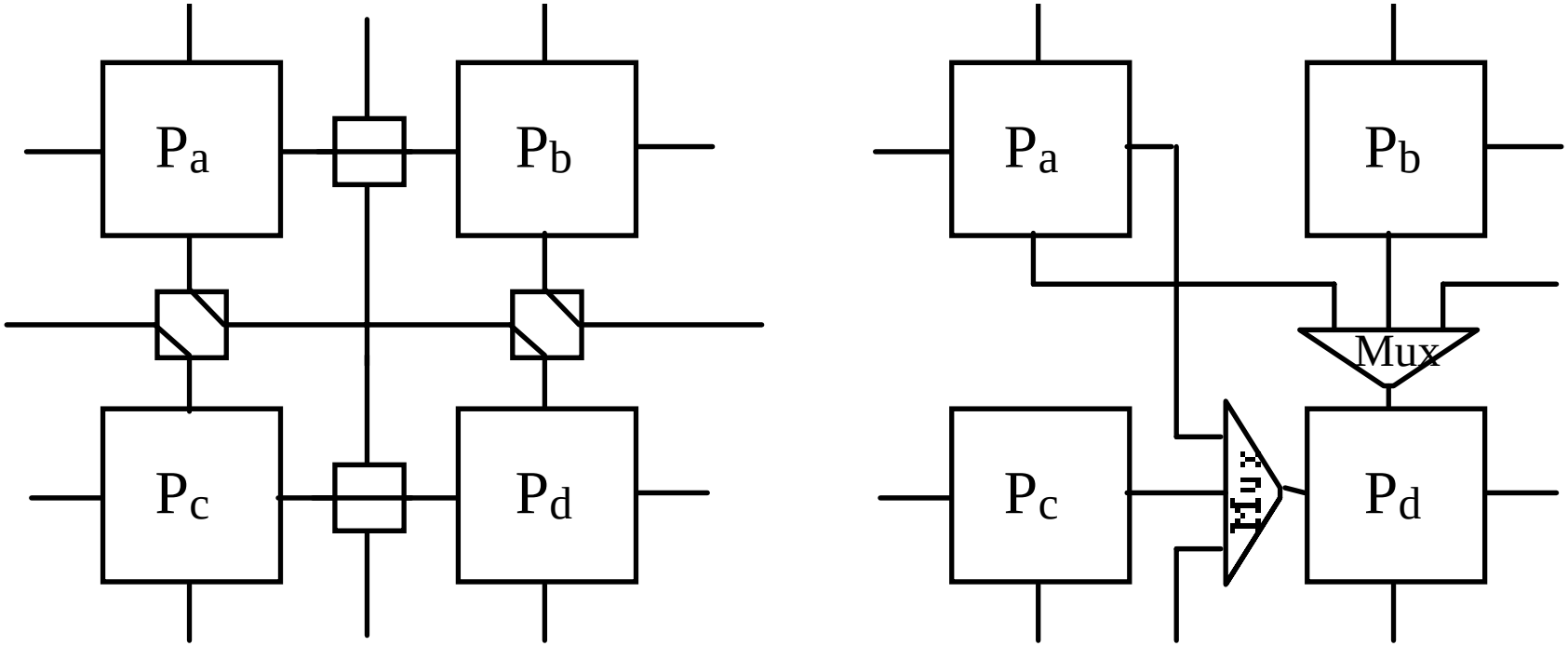


A linear array with a spare processor and reconfiguration switches



A linear array with a spare processor and embedded switching

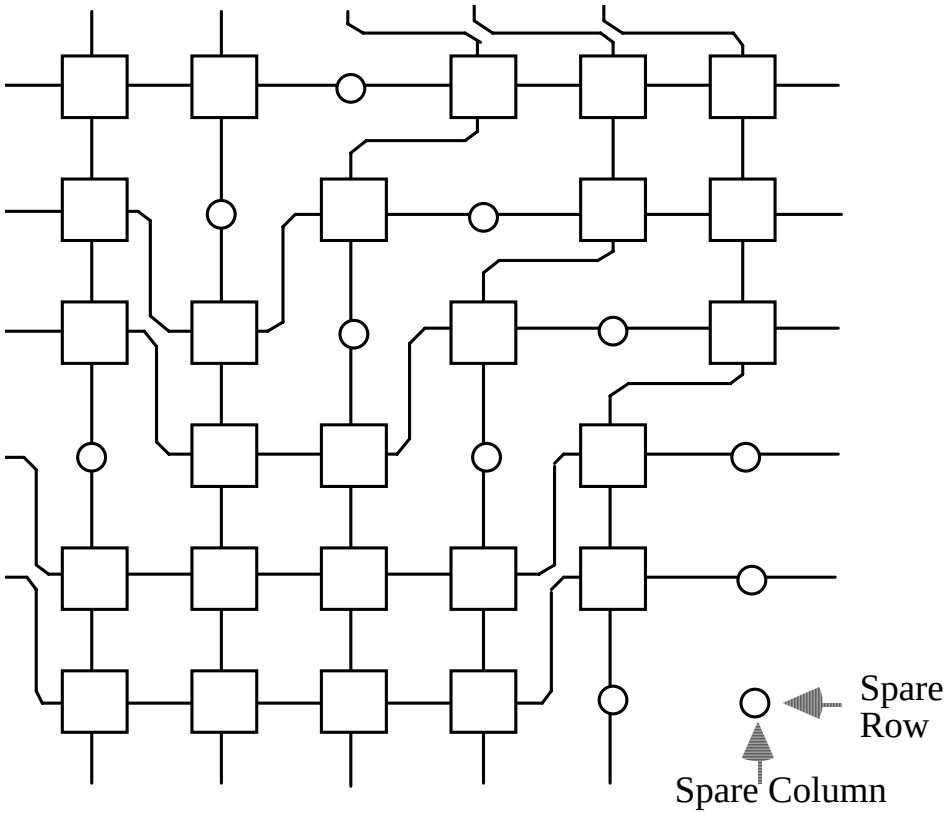
Defect Tolerance in 2D Arrays



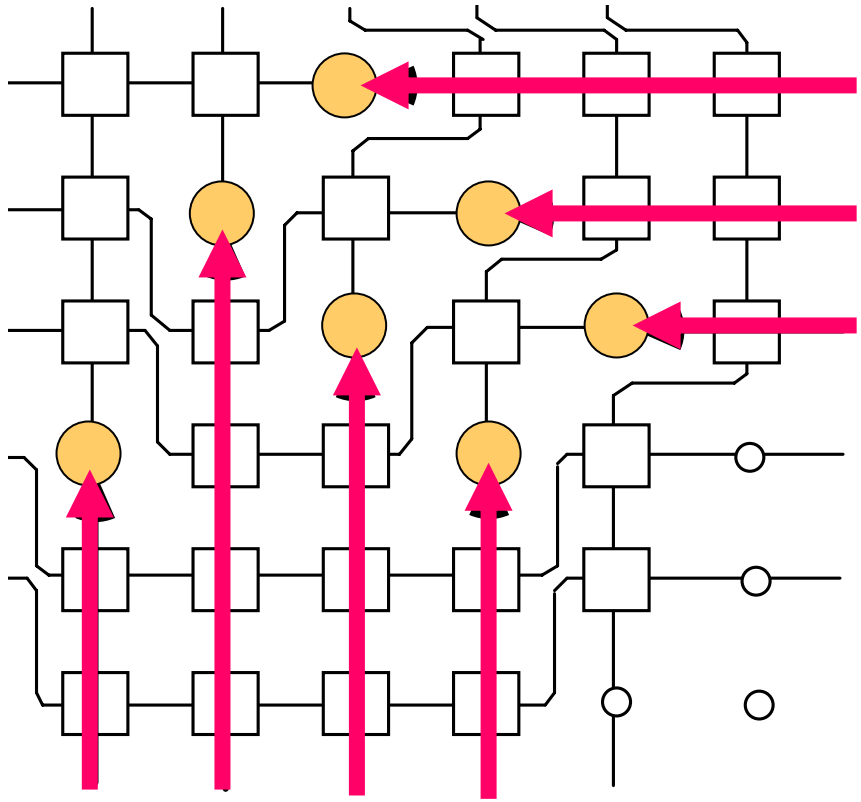
Two types of reconfiguration switching for 2D arrays

Assumption: A defective unit can be bypassed in its row/column by means of a separate switching mechanism (not shown)

A Reconfiguration Scheme for 2D Arrays



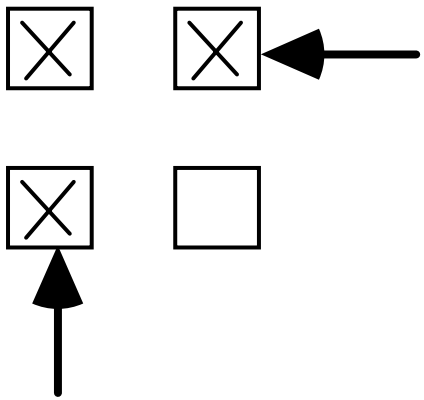
A 5 × 5 working array salvaged from a 6 × 6 redundant mesh through reconfiguration switching



Seven defective processors in a 5 × 5 array and their associated compensation paths

Limits of Reconfigurability

No compensation path
exists for this faulty node



A set of three defective nodes, one of which cannot be accommodated by the compensation-path method.

Extension: We can go beyond the 3-defect limit by providing spare rows on top and bottom and spare columns on either side