Infant Mortality Control

C. Glenn Shirley
Intel Corporation
Outline

- Introduction
- Manufacturing
- Methodology and Models
- Design for Infant Mortality Control
- Optimization of Infant Mortality Control
Introduction

• Silicon fabrication introduces latent reliability defects which cause early-life failure - infant mortality (IM).
• Without infant mortality control, some products have unacceptably high IM.
  – eg. Microprocessors need to have IM reduced from ~2000-5000 DPM in 0-30d to < 1000 DPM in 0-30d.
• We seek to control the “bathtub curve” perceived by customers by
  – Applying stress as part of manufacturing process flows.
    • Burn In to push weak units “over the edge” so that they can be screened in subsequent test.
  – Design for defect tolerance in “use”.
    • Hard defects appearing after test will not affect performance.
**Bathtub Curve**

**Failure Rate**

**Infant Mortality without Infant Mortality Control**

Indicator: Cumulative Fallout (DPM) or Fallout ÷ Interval (FITs) in interval.

- ~1 year
- 5-15 years

**Typical Fallout w/o IMC:** 3000 - 5000 DPM in 0-30d
Customer-Perceived Bathtub Curve

Infant Mortality with Infant Mortality Control

Indicator: Cumulative Fallout (DPM) or Fallout ÷ Interval (FITs) in interval.

Typical Goals: 100 - 1000 DPM 0-30d; 200 - 400 FITs 0-1y
Infant Mortality Control

• Manufacturing
  – Burn In units to activate latent reliability defects before final test.
  – Declining failure rate means that customer perceived IM is reduced.
  – Burn In conditions (time, temperature, voltage) are adjusted to meet IM and Wearout reliability goals, remain functional, and avoid thermal runaway.
  – Burn in power supply and thermal dissipation is becoming a big issue.

• Design
  – Design devices for tolerance to hard defects.
  – Fault tolerance design potentially impacts design costs, chip costs, and performance.
Outline

• Introduction

→ • Manufacturing
  • Methodology and Models
  • Design for Infant Mortality Control
  • Optimization of Infant Mortality Control
Manufacturing Flow

Wafer Fabrication
- Source of Si Fabrication Defects
- Density = D_fab

Assembly
- Source of Package Assembly Defects
- Opens/Shorts/Leakage

Use DPM From
- Test Holes
- Additional latent reliability defects: D_{use}

“Use”-Like Monitor

Improved Tests

Sort
- Initial functional screen at wafer level.
- Crude thermal & electrical environment
- “Loose” timings.
- Cold temperature to screen cold defects.

Burn In
- Exercise DUTs at Vcc and Tj > “use”.
- Limited by DUT power, and intrinsic rel degradation.
- Induces additional Si defect density D_{bi} (“turns on” latent rel defects.)

Post-Burn-In Test
- Opens/Shorts/Lkg = Assy Defects
- Functional failures correspond to Burn In-Induced Defects
- Fine speed screen at hot, low Vcc spec corner.

“Use”
Manufacturing Indicators & Controls

Defect Rel Model

Predicted Field Failure

Predicted IM at “Use”-Like Monitor

Predicted BI Fallout

IM Monitor

IM Failure

Test Hole DPM

t=0 Fallout

Assembly Failures
(Opens/Shorts) & Sort/Class Miscorrel’n

Si Functional Failures

“Use”-Like Monitor

Field Failure

Rejected Analysis

Predicted IM at “Use”-Like Monitor

“Use”-Like Monitor

“Use”

Reject Analysis

IM Monitor

IM Monitor

Reject Analysis

Field IM

t>0 Fallout

Yield Loss = Failed Dies/Total

Wafer Fabric’n  Sort  Assembly  Burn In  Post-Burn In Test

Infant Mortality Control, IRPS 2002

Page 9
Reject Analysis for “Use”-Like Monitor

Post Burn in Test

Update Test Pgm

"Use"-Like Monitor

Post Burn in Test (At QA conditions*)

Test Hole Candidate

Test Hole Resolution Methodology

* T, V to eliminate tester-tester miscorrel'n.

Analyze Signature

Test Process Error

IM Failure
Manufacturing Control of IM

- Reliability-related fallout after burn in is segregated from other fallout by reject analysis flows.
  - At final test (Rel Defect Monitor), and “Use”-like Monitor.
- Fallout predicted from Sort yield-loss via Defect Reliability Model is compared with actual fallout.
  - Excursions trigger corrective action
    - Possible root causes: Failure of BI hardware, Sort or Class test coverage issues, new failure mechanisms.
  - In-control monitors validate Defect Rel Model.
    - Rel Defect Model is used to tune burn in conditions using Goals.
- It is difficult to validate true field reliability failure rates.
  - Focus is on correlating mechanisms.
Power Management in BI

- Burn in is done at high Tj and Vcc, but low frequency.
  - Under these conditions, static power dominates. (Idyn is small.)
- Power has several contributions
  - $I_{total} = I_{sub} + I_{gate} + I_{dcap} + I_{dyn}$
  - $I_{sub}$ - subthreshold leakage current.
    - V-sensitive: increases 15-20% for a 0.1V increase
    - T-sensitive: increases 25-30% for a 10°C increase
    - Large (10X) within-wafer, -lot variation (sensitive to Le variation)
  - Oxide Leakage. Gate oxide leakage due to transistors ($I_{gate}$) and decoupling capacitors ($I_{dcap}$).
    - V-sensitive: increases 25-30% for a 0.1V increase
    - T-insensitive: increases 30% for an increase from 0°C to 95°C
    - tox-sensitive: increases 2.5x for a 1Å decrease
    - Small statistical variation.
Components of Burn in Power

- \( P_{\text{DCAP}} \)
- \( P_{\text{gate}} \)
- \( P_{\text{sub}} \)
- \( P_{\text{dyn}} \)

<table>
<thead>
<tr>
<th>Component</th>
<th>0.18 ( \mu )</th>
<th>0.13 ( \mu )</th>
<th>0.10 ( \mu )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P_{\text{DCAP}} )</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( P_{\text{gate}} )</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( P_{\text{sub}} )</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( P_{\text{dyn}} )</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Burn In Hardware Req’ts

- Variation in DUT leakage characteristics is reflected in Tj variation in the burn-in chamber.
- Ta must be set so that Tj for the hottest device cannot exceed reliability, functionality, and thermal runaway limits.
- Ta may be raised (reducing burn in time) by narrowing Tj distributions by
  - Improved (reduced) thermal impedances.
  - Slicing the Isb distributions based on Sort-measured Isb.
Air- vs Water-Cooled BI Hardware

But we can’t—too many hot units, too much thermal runaway.

Air-cooled

want to do this to minimize BIT

But we can’t—too many hot units, too much thermal runaway
Air- vs Water-Cooled BI Hardware

Improved thermal impedance gives shorter burn in times for the same Tjmax limit.
Outline

- Introduction
- Manufacturing
- **Methodology and Models**
- Design for Infant Mortality Control
- Optimization of Infant Mortality Control
Manufacturing
• BI TVF, time setpoints.
• BI Hardware Power/Thermal Characteristics

Process
• Reliability Characteristic
• Defect Characteristics
• Power Characteristics
• TVF Functional Limits

Product
• Power Management
• TVF Functionality limits
• Fault Tolerance

Use Condition Specs
• TVF Conditions

Optimizations done depend on stage in product lifecycle.

Defect Reliability Model
Scaling models of Area, Defect density, Acceleration, Use and BI TVF, etc.

Goals
• DPM/FIT Requirements

TVF = "Temperature, Voltage, Frequency"
Defect Reliability Models

- The Defect Reliability Model is critical to the control of burn in to meet customer IM requirements.
- The Defect Rel Model predicts IM reliability indicators as functions of
  - Sort Yield loss (fab defect density).
  - Defect reliability characteristics (rel statistics, acceleration).
  - Die size.
  - Product defect tolerance characteristics.
  - Burn in Time, Temperature, Voltage.
  - Useage Conditions (Temperature, Voltage).
- Models of Temperature and Voltage in Burn In and Use are inputs to Defect Rel Models.
  - Recent process generations require sophisticated models.
Extraction of IM “Baseline” Model

- The defect reliability of the Si process is characterized using SRAM data.
  - Probability time distribution is extracted.
  - T, V acceleration model is extracted.

- Defect reliability for Microprocessors is predicted from SRAM data, scaled for
  - Die Area, Fab defect density, Burn In Conditions, Use Conditions, defect/fault tolerant characteristics.

- Prediction is used to
  - Validate model vs “point check” Microprocessor life-test data.
  - Calculate burn in condition required to meet goals.
Data Collection for Baseline Model

- About 10k units are needed.
- Sort has a BI voltage test.
  - Test/Stress (< 1 sec)/Test
- Typical BI readouts 3, 6, 12, 24, 48, 168h with extended stress to 1kh.
- Establish reliability distribution at burn in T,V.
- Determine acceleration by branch at lower T,V.
  - Sequential stress can reduce device hour requirements.
SRAM Baseline (.25μ Technology)

- Lognormal, voltage-accelerated model was fitted to lifetest data at multiple voltages
- \[ TTF \propto e^{-C \cdot V} \]
  - \[ C = 7.0 \pm 1.4 \]
- Acceleration from normal burn-in voltage (2.5V) to normal operation (1.8V) is about 130x

Source: Neal Mielke
### SRAM & Microprocessor Life Test Data

<table>
<thead>
<tr>
<th></th>
<th>RO</th>
<th>6</th>
<th>24</th>
<th>48</th>
<th>168</th>
<th>500</th>
<th>1000</th>
<th>2000</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRAM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>8</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>SS</td>
<td>2460</td>
<td>2451</td>
<td>2448</td>
<td>2445</td>
<td>936</td>
<td>698</td>
<td>461</td>
</tr>
<tr>
<td>Microprocessor</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RO</td>
<td>6</td>
<td>24</td>
<td>48</td>
<td>168</td>
<td>500</td>
<td>1000</td>
<td>2000</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>13</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>SS</td>
<td>2865</td>
<td>2852</td>
<td>1377</td>
<td>741</td>
<td>372</td>
<td>173</td>
<td>79</td>
</tr>
</tbody>
</table>

- **RO**: Readout hours (or cycles, etc.)
- **F**: Number of failures at the readout
- **SS**: Sample size at the readout
- **0.35μ technology**
Lognormal Reliability Distribution

• Fit failures in time to a lognormal distribution in time.
  \[ F(t) = \Phi\left(\frac{\ln t - \mu}{\sigma}\right) \]
• \(\mu\) defines the median time-to-fail.
  \[ t_{50} = \exp(\mu) \]
• \(\sigma\) defines the shape
  – Large \(\sigma (> 2)\) means high early failure rate decreasing with time.
  – Small \(\sigma (< 0.5)\) means increasing (wearout) type of failure.
  – \(\sigma\) near 1 means roughly constant failure rate.
• \(\Phi(z)\) is the normal probability function.
  \[ \Phi(z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{z} e^{-z'^{2}/2} \, dz' \]
Extraction of SRAM Baseline from Life-Test Data

- Plot cumulative % fail vs. time
  - Probability plot vs. log t

- Determine $\mu$ and $\sigma$
  - Plot $\ln(t_i)$ on y axis*
  - Plot $\Phi^{-1}(F_i)$ on x-axis
  - Slope is $\sigma$
  - Intercept is $\mu$

* Differs from orientation of graph shown.
Acceleration Factor

- Subject the same population to two different stress tests:
  - Low Stress Test 1: Low temperature $T_1$, low voltage $V_1$. In time interval $t_1$, a certain proportion, $X$, fails.
  - High Stress Test 2: High temperature $T_2$, high voltage $V_2$. It takes a (shorter) time interval $t_2$ for the same proportion, $X$, to fail.

- The acceleration, greater than 1, of case 2 relative to case 1 is $A = t_1/t_2$.

- In general acceleration is the ratio of times for the “same effect”.
  - Think of a clock at running at different rates depending on the temperature and voltage of a stress test.
Acceleration Factor ct’d

• We determine a cumulative distribution function at a high stress condition (usually high voltage and high temperature): \( F_2(t) \)

• What is the cumulative distribution function, denoted by \( F_1(t) \) at a different condition 1?

\[
F_1(t) = F_2\left( \frac{t}{A_{21}} \right)
\]

• The same scaling applies to \( S \):

\[
S_1(t) = S_2\left( \frac{t}{A_{21}} \right)
\]
Acceleration Factor ct’d

- We use the Arrhenius Model for temperature acceleration + voltage acceleration:

\[ A_{21} = \exp\left\{ \frac{Q}{k} \left[ \frac{1}{T_1} - \frac{1}{T_2} \right] + C(V_2 - V_1) \right\} \]

- \( T_2, V_2, T_1, V_1 \) are operating temperatures (in deg K) and voltages at conditions 2 and 1, respectively.
- \( k = 8.61 \times 10^{-5} \text{ eV/K} \) is Boltzmann’s constant.
- \( Q \) (eV) is the thermal activation energy
- \( C \) (volts\(^{-1}\)) is the voltage acceleration constant.
Acceleration Example

• For the SRAM example, burn-in data were acquired at $T_j = 135^\circ C$ and 4.6V.
• What are the cum. fail distribution at use conditions ($T_j = 85^\circ C$, 3.3V)?
• Acceleration between use and burn-in is 317.3 (assuming $Q = 0.6$ eV, $C = 2.6$ volts$^{-1}$).

$$F(t) = \Phi\left(\frac{\ln(t/317.3) - 71.02}{25.73}\right)\text{(SRAM)}$$

Argument of log function is time at condition that model was fitted to data. Use-condition clock runs 317.3 times slower.
Acceleration Example ct’d

Distribution shifts right for deceleration, left for acceleration.

Lognormal with two-sided 90.0% confidence limits
Burn-In Example

• SRAM is burned in for three hours; what is its use survival function?

• Fraction of pre-burn-in unstressed population surviving is

\[ S(t) = 1 - F(t) = 1 - \Phi\left(\frac{\ln(3 + t/317.3) - 71.02}{25.73}\right) \]
Burn-In Example continued

• Proportion surviving seen by the customer is

\[
S_{Use}(t) = \frac{1 - \Phi \left( \frac{\ln(3 + t/3173) - 71.02}{25.73} \right)}{1 - \Phi \left( \frac{\ln(3) - 71.02}{25.73} \right)}
\]

Exact

Normalize so that customer’s proportion surviving at his \( t = 0 \) is 1

• For small fallout (< 5%, say) this approximates to

\[
F_{Use}(t) = \Phi \left( \frac{\ln(3 + t/3173) - 71.02}{25.73} \right) - \Phi \left( \frac{\ln(3) - 71.02}{25.73} \right)
\]

Approximate
Effect of burn-in:
Greatest at early times.

3 hours of Burn-in
removes about 3x317.3
hours worth of defects

Lognormal with two-sided 90.0% confidence limits
Reliability Indicator Examples

• Reliability indicators can be expressed in terms of the survival function at use conditions after burn in, $S(t)$.

• Formulas
  
  – Fraction failing between two times, $t_1$ and $t_2$. 
  
  \[ S(t_1) - S(t_2) \]

  – Average failure rate between two times, $t_1$ and $t_2$. 

  \[ \frac{\ln S(t_2) - \ln S(t_1)}{t_1 - t_2} \]

• Examples

  – 0-30d DPM

  \[ 10^6 \times \{1 - S(t = 720\text{hours})\} \]

  – 0-1y average failure rate in FITs.

  \[ 10^9 \times \ln[S(t = 8760\text{hours})] / 8760 \]
## Failure Rate Units

### Equivalent failure rates in different units:

<table>
<thead>
<tr>
<th>Fraction failing per hour</th>
<th>% failing per 1Khr</th>
<th>FIT</th>
<th>DPM in 0-1yr</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.000001</td>
<td>1.0</td>
<td>10,000</td>
<td>87600</td>
</tr>
<tr>
<td>0.000001</td>
<td>0.1</td>
<td>1,000</td>
<td>8760</td>
</tr>
<tr>
<td>0.0000001</td>
<td>0.01</td>
<td>100</td>
<td>876</td>
</tr>
<tr>
<td>0.00000001</td>
<td>0.001</td>
<td>10</td>
<td>88</td>
</tr>
<tr>
<td>0.000000001</td>
<td>0.0001</td>
<td>1</td>
<td>9</td>
</tr>
</tbody>
</table>

### Conversion factors:

- Failures per hour $\times 10^5 = \%$ per Khr
- Failures per hour $\times 10^9 = \text{FIT}$
- $\%$ per Khr $\times 10^4 = \text{FIT}$
- $\text{FIT} \times 8760\text{hrs} \times 10^6 \text{DPM} \div 10^9 \text{FIT} = 0-1\text{yr DPM}$
Burn-in has the biggest effect on early early-life indicators, eg (0-30 day DPM) vs 0-1 year FIT.

Determine Burn In Time
Reliability Modeling Summary So Far

• Account of acceleration, by modifying the time argument of the fitted distribution by dividing by the acceleration.
  – As if the rate of the clock depends on T, V.

• To take account of burn in:
  – Account for the stress history in the time argument of the fitted distribution.
  – Normalize the survival function to be unity at the customer’s t = 0.

• Acceleration and burn-in effects are taken account of in convenient formulae for indicators.

• We still need to cover scaling functions for (i) defect density, (ii) area, (iii) fault tolerance.
Defect Reliability

- We now specialize the reliability models to models of defect reliability to get defect density, and area scaling.
- Infant Mortality reliability is driven by defects.
- Defects from the same source affect both yield and infant mortality.
  - Yield is fallout measured before any stress.
    - Contributions come from Sort (wafer-level functional test) and pre-burn-in class test.
    - Depends on “yield defect density”, Dyield. (Kill devices at t=0.)
  - Infant mortality is measured by fallout due to stress
    - Largely post-burn-in class test, but Sort stress tests too.
    - Depends on “reliability defect density”, Drel. (Kill devices for t > 0.)
Defect-On-Grid Model

- OK, never a yield or reliability issue.
- Sometimes a latent reliability defect, sometimes OK.
- Sometimes a yield defect, sometimes a latent reliability defect, sometimes OK.
- Always a yield defect.

Latent Reliability Defect

Either:
Particle does not touch conductors, but both sides are within $\delta$ of the conductor.

or:
Particle touches one conductor and is within $\delta$ of its neighbor.
Concept of Reliability Defect Density

\[
D_{rel} = \text{Constant} \times D_{yield}
\]

Assumption of Model: Proportional because...
Both kinds of defects are from the same source.
Models of Defect Density

- Latent reliability defects affecting burn in and “use” come from the same source as defects which affect Sort yield.
  - Paretos match.
  - Latent rel. defect density is ~ 1% of Sort defect density.

\[ F(t) = \text{Proportion failing at time } t \]
\[ D_{\text{rel}} = -\ln\{1 - F(t = x \text{ h})\} / \text{Area} \]
\[ D_{\text{yield}} = -\ln\{\text{No of Good Die} / \text{Total No of Die}\} / \text{Area} \]

Scaling Concept for Defect Reliability

- Each latent reliability defect has a “lifetime”.
  - Collectively described by a defect survival probability, $s(t)$.
- Die survival probability, $S(t)$, is the product of defect survival probabilities.
  - Assumes randomly distributed noninteracting defects (“Poisson statistics”)
- Density of latent reliability defects is $D_{rel}$ (cm$^{-2}$), and die area is $A$.
- If the first “activation” of a latent reliability defect is fatal to the die (no functional redundancy), then $S(t)$ is a product of $s(t)$’s for defects.
  - We’ll extend this to fault tolerant circuits later.
Scaling Concept for Defect Reliability

$$S(t) = [s(t)]^{Number\ of\ Latent\ Rel.\ Defects\ on\ Die} = [s(t)]^{D_{rel} \times A}$$

$$S'(t) = [s(t)]^{2 \times AD_{rel}}$$

$$S'(t) = S(t)^2$$

Double Area

Double Rel Defect Density

Double the defect density, or double the area = Square the survival function.
Scaling Concept for Defect Reliability

• This suggests a defect density and die area scaling law for the die survival function.

\[
S'(t) = S(t) \left( \frac{D'_{\text{rel}} \times A'}{D_{\text{rel}} \times A} \right) \left( \frac{D'_{\text{yield}} \times A'}{D_{\text{yield}} \times A} \right)
\]

• Depends on observed correlation between Yield and Reliability Defect Densities.
• Yield defect density is 100x larger than rel defect density and can be measured at Sort.
Example: Area Scaling of Defect Rel

- A useful approximation to

\[ S'(t) = S(t) \left( \frac{D'_\text{yield} \times A'}{D_{\text{yield}} \times A} \right) \]

is

\[ F'(t) = \frac{D'_\text{yield} \times A'}{D_{\text{yield}} \times A} \times F(t) \]

- For the SRAM/microprocessor example

\[
F_{\text{Microprocessor}} = \frac{D_{\text{Microprocessor}} \times A_{\text{Microprocessor}}}{D_{\text{SRAM}} \times A_{\text{SRAM}}} \times F_{\text{SRAM}}
\]

\[
= \frac{(D_{\text{Microprocessor}} \approx D_{\text{SRAM}}) \times 378 \text{ mils} \times 348 \text{ mils}}{D_{\text{SRAM}} \times 284 \text{ mils} \times 295 \text{ mils}}
\]

\[
= 1.45 \times F_{\text{SRAM}}
\]
Example: Area Scaling of Defect Rel

Fit to microprocessor data (Red) = 1.45 × SRAM Model (Blue)

SRAM Data  Microprocessor Data

SRAM Model (Least-squares fit)
Distribution Scaling

- Normal Probability Scale
- Logarithm of Time
- Increase Yield Defect Density
  - Increase Area
- Deceleration and Use conditions
- Decrease Yield Defect Density
  - Decrease Area
- Acceleration
- Burn-In
Outline

• Introduction
• Manufacturing
• Methodology and Models

→ • Design for Infant Mortality Control
• Optimization of Infant Mortality Control
Design for Infant Mortality Control

- Burn In reduces the number of latent reliability defects escaping final test.
- An alternative approach is to make dies tolerant to hard defects in “use”.
- We derive a simple model which shows the infant mortality DPM benefit of “hard” fault tolerance.
- Manufacturing benefits derive from
  - Reduced burn in time.
  - Lower power requirements if areas of dies “immune” to hard defects don’t need to be powered in burn in.
Models of Defect Density, ct’d

- Latent Reliability Defect Density vs Time & Stress
  - Lognormal time cumulative fraction failing distribution is used.
  - $\sigma$, $\mu$, and AF are determined from test chip (SRAM) post-burn in test fallout vs burn in time and Tj, Vcc variation experiments.
  - Example values: $\sigma = 25$, $\mu = 70$, AF = 200.

$$D_{\text{fab}} \approx 0.01 \times D_{\text{fab}}$$
$$D_{\text{bi}} \approx 0.01 \times D_{\text{fab}} \frac{\ln [1 - \Phi \left( \frac{\ln(t_{bi}) - \mu}{\sigma} \right)]}{\ln [1 - \Phi \left( \frac{\ln(t \text{ h}) - \mu}{\sigma} \right)]}$$
$$D_{\text{ase}} \approx 0.01 \times D_{\text{fab}} \frac{\ln [1 - \Phi \left( \frac{\ln(t_{bi} + t_{use} / \text{AF}) - \mu}{\sigma} \right)]}{\ln [1 - \Phi \left( \frac{\ln(t \text{ h}) - \mu}{\sigma} \right)]}$$

(Assumes that the BI defect density is defined at 1h of BI.)
Redundancy Statistics

• Chip has repairable (usually cache) and non-repairable (usually random logic) areas.
  – Define $r = \frac{A_{\text{repairable}}}{A_{\text{total}}}$

• The repairable area of the chip is divided into a number “n” of repairable elements.
  – The larger n is, the more “survivable” is the chip, and the greater is the design/area overhead.

• Each repairable element is characterized by the number of defects it can “survive”.
  – Assumption here: Repairable elements can survive up to 1 defect, and non-repairable cannot survive more than 0 defects.
  – There are different circuit/logic ways to realize this.

Note: This description is an approximation intended only to show the major sensitivities.
Redundancy Statistics, cont’d

• Some *kinds* of defects are fatal even to repairable elements, depending on the redundancy scheme used.
  – $f =$ fraction of all kinds of defects which can be repaired by repairable elements.

Two Limiting Special Cases

- No redundancy at all. ($f \times r = 0$, irrespective of $n$). Yield and Infant Mortality for $A_{total}$.
- Ideal Redundancy. ($n =$ very large). Yield and Infant Mortality for $A_{non-repairable}$
Yield Example

- Test programs at first test screen (eg. Sort) detect faults and connect “spare” elements (eg. by fusing).
  - Big yield gain for $n = 1$, diminishing return for $n > 1$.

![Graph showing Yield % vs Area x Defect Density](image)

- "Ideal Redundancy": 80% of area and/or defects is/are repaired.
- Area and/or repairable defect fraction, $f \times r = 0.8$
Redundancy Model for Yield

• Probability of a good die after Sort is given by

\[
Y = [Y_r^0 + Y_r^1]^n Y_{nr}
\]

• Using Poisson expressions for probabilities in terms of defect density we get

\[
Y = \left(1 + \frac{f \times r \times A_{tot} \times D}{n}\right)^n \times \exp(-A_{tot} \times D)
\]
Infant Mortality & Fault Tolerance

- Main opportunity is “in use” repair of latent reliability defects escaping burn in - “Infant Mortality”.
  - Very little gain in yield for repair after burn in.
- Requires on-chip logic to detect and replace failing elements with “spares”, or correct data in failing elements.
- What is fraction of dies failing in 0-30d which have survived Sort, burn-in, and post burn-in test?
  - Account for repairs at Sort making redundant elements unavailable at burn in and in “use”.
  - As function of $f$, $r$, $n$, and burn in time ($t_{bi}$)

Note: The following examples are not representative of Intel processes.
Infant Mortality Large Die Example

- 16-elements are needed to get most of available benefit.
- 10-20X burn in time reduction, depending on goal.

"Ideal Redundancy": 80% of die is repaired.

Area = 4 cm²
Area and/or repairable defect fraction, f x r = 0.8
Infant Mortality Small Die Example

- 1 redundant element is sufficient for a large effect.
- Burn In stress time may be reduced enough to move the stress to a test socket. (10^{-3} h = 3.6 sec).

![Graph showing the relationship between Burn In Time (h) and DPM (Defects Per Million) for different redundancy levels.](image)

- **No Redundancy**
- **n = 1**
- **n = 2**

**Ideal Redundancy**: 80% of die is repaired.

**Area = 1 cm^2**

**Area and/or repairable defect fraction, f \times r = .8**
Infant Mortality & Redundancy, c’td

- The customer-observed fraction surviving burn in plus “use”, is:

\[
U = \left[1 + \frac{f \times r}{n} \frac{A_{total}(D + D_{use})}{A_{total}(D + D_{bi})}\right]^n \times \exp\left[-A_{total} \times (D_{use} - D_{bi})\right]
\]

where Poisson probability functions in terms of defect density were used.

- So Infant Mortality DPM after \( t_{use} = 720 \, h/30 \, d \) and after \( t_{bi} \) of burn in is

  Infant Mortality DPM = \( 10^6 \times (1 - U) \)
Fault-Tolerance Requirements

• Infant Mortality benefit requires “In Use” fault tolerance.
  – Mostly cache-oriented on-chip schemes, transparent to OEMs.

• Fault-tolerance requires:
  – Test to detect faults.
  – Logic to replace failing elements with “spares”, or to correct data.

• Kinds of In-Use Fault Tolerance
  – Test during POST, set up logic to avoid faults (redundancy).
    • Doesn’t reliably cover all spec conditions.
  – On-the-fly fault detection and repair/correction (ECC).

• Optimal implementation depends on
  – Effectiveness. Kind of scheme vs kind of defect vs defect pareto.
  – Cost: Area impact.
  – Performance impact.
Kinds of Repair Schemes

Block Repair

Column Repair

One bit in a row defective

Defective bit, corrected by ECC

Source: Ben Eapen
4 Major failure modes in cache

- Random Single-Bit Fails predominate.
- Clustered (in Row/Column) Single Bit Fails
- Column Fails
- Row Fails
- Array Fails

Source: Ben Eapen
Repair Efficiency

<table>
<thead>
<tr>
<th>Repair Scheme</th>
<th>Block</th>
<th>Column</th>
<th>Row</th>
<th>ECC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random SB</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Clustered SB</td>
<td>✔</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Column</td>
<td>✔</td>
<td>✔</td>
<td>✗</td>
<td>-</td>
</tr>
<tr>
<td>Row</td>
<td>✔</td>
<td>✗</td>
<td>✔</td>
<td>✗</td>
</tr>
<tr>
<td>Array</td>
<td>✔</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
</tr>
</tbody>
</table>

**Source:** Ben Eapen

- **H/M/L = High/Med/Low**
- f is large (~1)
- f is small (~0)
- f depends in details of pareto & implementation
Outline

• Introduction
• Manufacturing
• Methodology and Models
• Design for Infant Mortality Control
  • Optimization of Infant Mortality Control
Optimization of Infant Mortality Control

• Control Defect Characteristics
  – Reduce density, especially of low acceleration defects.

• More Precise Definition of Use Conditions
  – Determined by performance requirements.
  – Segment products by “use” condition.
  – More accurate models of “use” conditions vs guardband by worst-case.

• Make circuits tolerant to hard defects.
  – Cache is the best opportunity.
    • For microprocessors, a trend is towards large dies having lots of cache.
  – Design requirements may impact performance and area.
Optimization of Infant Mortality Control

• Increase BI Conditions to fundamental limits
  – Intrinsic reliability of oxides, etc.
  – Functionality of circuits at TVF corner required for toggle coverage is a compromise with performance.

• Improve thermal/power control in burn in.
  – Design products with power management on die
    • eg power down cache if it is hard-fault-tolerant and does not need to be burned in.
    • eg. sequential power of die subareas can fit dies into equipment envelope, but extends burn in times.
  – Lower thermal impedances in burn in hardware to reduce thermal runaway and make Tj distributions narrower.
    • Higher median temperatures with hottest units still in thermal control reduces burn in time.