A Cost Analysis Framework for Multi-core Systems with Spares

Saeed Shamshiri, Peter Lisherness, Sung-Jui Pan, and Kwang-Ting Cheng
Department of Electrical and Computer Engineering
University of California, Santa Barbara, CA 93106
{saeeedshamshiri, peter, srpan, timcheng}@ece.ucsb.edu

Abstract

It becomes increasingly difficult to achieve a high manufacturing yield for multi-core chips due to larger chip sizes, higher device densities, and greater failure rates. By adding a limited number of spare cores to replace defective cores either before shipment or in the field, the effective yield of the chip and its overall cost can be significantly improved. In this paper, we propose a yield and cost analysis framework to better understand the dependency of a multi-core chip’s cost on key parameters such as the number of cores and spares, core yield, and defect coverage of manufacturing and in-field testing. Our analysis shows that we can eliminate the burn-in process when we have some spare cores for in-field recovery. We demonstrate that a high defect coverage for in-field testing, a necessity for supporting in-field recovery, is essential for overall cost reduction. We also illustrate that, with in-field recovery capability, the reliance on high quality manufacturing testing is significantly reduced.

1. Introduction

Multi-core processors and SoCs are quickly becoming the dominant architecture, driven by shrinking processes and diminishing returns from additional per-core complexity [1-5]. While the individual core complexity is tapering off, however, the larger die size, increasing device counts, and higher defect rates are leading to lower yields [6,7].

A common approach to solve this yield problem has been the addition of redundancy [8]. This approach has become ubiquitous in memories, whose highly repetitive structures and high densities both enable and demand a degree of defect tolerance [9,10]. It has also been proposed for use in logic devices with similarly repetitive structures, such as programmable logic arrays (PLAs) [8], field programmable gate arrays (FPGAs) [11], and systolic array processors [12]. More recently, it has been used in single-chip multiprocessors [13].

In this paper we present a yield and cost analysis framework for multi-core processors that uses a spare scheme to recover both manufacturing and in-field failures. We model yield, manufacturing cost, and service cost, and consider the burn-in process in our analysis. Some previous works have addressed this issue [6,8], but none have included burn-in in their analysis. In our model, we consider factors such as the raw yield of cores, the defect coverage of the manufacturing testing, and the defect coverage of the in-field testing. The defect coverage of manufacturing testing can be very different from that of in-field testing because these operate under different constraints, so we have distinguished between them in our model. Our experimental results show that with good in-field testing and sufficient spares for recovery, we can loosen manufacturing test requirements and eliminate the burn-in process as well.

Many papers have already shown that high quality in-field testing is feasible, either by built-in self-test or online testing [14-18]. Many of these approaches are software-based, incurring negligible area overhead, so they do not significantly add to the cost of the device. Our results assume the use of these low-overhead testing techniques for the sake of simplicity, but the analysis framework is still applicable when hardware-based testing and other types of online checking schemes are employed. The spare processors may even serve as a hardware checking device in a roving emulation scheme, as discussed in [19].

Previous studies have advocated the elimination of burn-in, particularly replacing it with high-stress $I_{DDQ}$ (quiescent supply current) testing [20-22]. In [20], the authors analyze the benefits resulting from burn-in elimination and conclude that the elimination of the substantial burn-in cost, reportedly 5% to 40% of the chip sale price, overwhelms the increased service cost from test escapes and $I_{DDQ}$ test cost. Unfortunately, this technique becomes less useful for nanometer-scale devices, where high intrinsic $I_{DDQ}$ makes it difficult to identify defective chips [23]. One of the contributions of this paper is an evaluation framework for the feasibility of applying existing defect tolerance methods to eliminate burn-in.

The rest of the paper is organized as follows: In the next section we present a core yield model, which forms the basis for further analysis. In Section 3, we expand this core yield into a yield model for an $m$-out-of-$n$ core system, where $n$ cores are manufactured but only $m$ are required for proper system operation. We then introduce a cost model for this system in Section 4, where we establish the manufacturing and service costs for these systems and show how each is affected by burn-in. In Section 5, we present experimental results, where the model is used to analyze the effect of additional spares, core yield, test
quality, and burn-in on the total system cost. Section 6 contains a brief summary and conclusion.

2. Core Yield Model

In this paper, a negative binomial yield model [7,24,25] is used to approximate the core yield. It is denoted as \( y_c \) and is a function of the defect density (\( \lambda \)), core area (\( A \)), and clustering parameter (\( \alpha \)). The clustering parameter determines the degree to which the defects are clustered. The resulting model is:

\[
y_c(A, \lambda, \alpha) = \left(1 + \frac{\lambda \times A}{\alpha} \right)^{-\alpha}
\]

(1)

Note that, \( \alpha \rightarrow 0 \) corresponds to a stronger clustering of defects, so a greater yield would be expected. On the other hand, \( \alpha \rightarrow \infty \) corresponds to weaker defect clustering, resulting in a lower yield. The range of \( \alpha \) for current technologies is between 0.3 and 5.0 [25].

Equation (1) shows the true yield of the core, but not the observed yield. The observed yield is greater than the true yield due to the imperfect testing process. Ideally, if the defect coverage of the manufacturing test is 100%, then observed yield is equal to the true yield. The observed core yield is the probability of a core passing the manufacturing test, which can be modeled as:

\[
y'_c(A, \lambda, \alpha, \Omega) = \left(1 + \frac{\lambda \times A \times \Omega}{\alpha} \right)^{-\alpha}
\]

(2)

Here, \( \Omega \) denotes the defect coverage of the manufacturing test for a single core. If the defect coverage \( \Omega \) is equal to zero, \( y'_c \) is equal to one, meaning that all cores pass the test because no defects are detected. On the other hand, if the defect coverage \( \Omega \) is equal to one, only a defect-free core can pass the test, and the observed yield (\( y'_c \)) is equal to the true yield (\( y_c \)).

3. Multi-Core System Yield Model

We assume that the system consists of a total number of \( n \) identical cores. These include \( m \) cores required for functional operation, which are called active cores for the rest of the paper, and \( n-m \) spare cores. If any one of these \( m \) active cores is detected as faulty during either manufacturing testing or in-field testing, the system can replace it with one of the spares. This increases yield and allows elimination of the expensive burn-in process by relying on the fault-free spares to replace failed cores during the warranty period.

After manufacturing, we test all \( n \) cores. Among those that pass the manufacturing test, we choose \( m \) cores as active cores. The observed system yield is the chance that at least \( m \) cores pass the test:

\[
y'_{sys} = \sum_{i=m}^{n} \binom{n}{i} (1-y'_c)^{n-i} (y'_c)^i
\]

(3)

The system will function when these \( m \) cores are indeed defect-free. The probability that a core is defect-free given that it passes the test is:

\[
q_c = \frac{y'_c}{y'_c'}
\]

So the probability that all \( m \) active cores are indeed defect-free after they all passed the test is:

\[
q_{sys} = q_c^m = \left(\frac{y'_c}{y'_c'}\right)^m
\]

(5)

Therefore, the true system yield can be expressed as:

\[
y_{sys} = y'_{sys} \times q_{sys}
\]

(6)

In our analysis, we assume that each individual core’s fault-free (or defect-free) probability is independent of that of other cores. While this assumption might not be always true, it gives us a first-order approximation and makes the analysis more feasible.

4. Chip Cost Model

If the costly burn-in process is skipped, the manufacturing test cost could be reduced. However, the chip would have a greater probability of failure within the warranty period and may incur a significantly greater service cost due to replacement.

On the other hand, a multi-core system with spares would incur a reduced service cost because a core that fails in the infant mortality period can be replaced by a spare. In the following, we evaluate the trade-offs of eliminating the burn-in process for a multi-core system with spares, based on a chip cost model that can be used to determine the number of required working spares in a shipped chip for overall cost minimization.

The cost of a shipped chip throughout its lifecycle, denoted as \( C_{Total} \), consists of two components – the manufacturing cost, \( C_{Man} \), and the service cost, \( C_{Ser} \):

\[
C_{Total} = C_{Man} + C_{Ser}
\]

(7)

In the next two sub-sections, we discuss the manufacturing and service cost model assuming burn-in is not employed. Then, in Section 4.3, we add burn-in to our model and discuss its effect on manufacturing and service costs.

4.1 Manufacturing Cost Model

The manufacturing cost per shipped chip, \( C_{Man} \), can be modeled as:

\[
C_{Man} = n \times C_K \times y'_{sys}
\]

(8)

Where \( n \), \( C_K \), and \( y'_{sys} \) denote the number of cores per chip, manufacturing cost per core, and the observed system yield of the chip, respectively. Notice that all chips that pass the...
manufacturing test will be sold in the market whether they truly work or not, as implied by the use of the variable $y'_{sys}$ instead of $y_{sys}$ in (8). The test escapes, which are proportional to the difference of $y'_{sys}$ and $y_{sys}$, will be accounted for in the service cost formulas.

Equation (8) indicates that the number of spares can affect the $C_{Man}$ in two ways: (1) adding more spares to a chip (i.e. a greater $n$) results in a bigger die and increases the per-chip manufacturing cost; (2) having more spares increases the chance of successful repair, resulting in a greater observed system yield, $y'_{sys}$, and decreasing the manufacturing cost per shipped chip. Our analytical results in Section 5 will illustrate that there is an optimal value for the number of spares such that the latter effect is more significant than the former effect, resulting in minimal manufacturing cost.

### 4.2 Service Cost Model

When a chip fails in the field within the warranty period, it incurs a service cost, potentially including the replacement of the entire system containing the chip and field technician labor. We model the expected service cost per shipped chip as:

$$ C_{Ser} = (1 - R_F)C_F $$

Where $C_F$ is the service cost per failed chip, and $R_F$ is the probability of a shipped chip not failing within the warranty period. The number of spares, $s$, in a shipped chip ranges from 0 to $n-m$. Chips with more spares have a greater chance of surviving the warranty period. Therefore, $R_F$ can be modeled as:

$$ R_F = \sum_{s=0}^{n-m} P(s) \times F(s, t) $$

Where $P(s)$ denotes the probability of a shipped chip having $s$ spares, and $F(s, t)$ denotes the probability that a chip with $s$ spares does not fail within the time $t$ after shipment. We need to model these two parameters to be able to compute the service cost.

$P(s)$, defined above, is equal to the probability of a manufactured chip having $s$ spares (denoted as $R(s)$) divided by the probability of shipping a manufactured chip (i.e. observed system yield):

$$ P(s) = \frac{R(s)}{y'_{sys}} $$

Further, the probability that an $m$-out-of-$n$-core system has exactly $s$ spares (i.e. $R(s)$), can be modeled as:

$$ R(s) = \binom{n}{m+s}(1-y'_c)^{n-m-s}(y'_c)^{m+s} $$

$R(s)$ is the $(s+1)$-th term in Equation (3), so we can rewrite Equation (3) based on $R(s)$ as follows:

$$ y'_s = \sum_{s=0}^{n-m} R(s) $$

As the last step of computing the service cost, we need to model $F(s, t)$. To do that, first, we have to look at the failure rate in the lifecycle of a core. Figure 1 shows the well-known bathtub curve indicating that the field failure rate throughout a core’s lifecycle has three distinct periods: infant mortality, grace, and breakdown. The failure rates in these periods can be characterized by the Weibull distribution model [26], which has two parameters: a shape ($\beta$) and a scale parameter ($\mu$). As a result, the failure rate can be modeled as:

$$ f(t, \mu, \beta) = \frac{\beta}{\mu} \left(\frac{t}{\mu}\right)^{\beta-1} $$

The scale parameter is defined as the time in which 63.2% of failures occur. A greater $\mu$ indicates a longer interval between failures. If $\beta$ is less than one, the failure rate decreases with time (i.e. the infant mortality period); if $\beta$ is equal to one, the failure rate is a constant (i.e. the grace period); and, finally, if $\beta$ is greater than one, the failure rate increases with time (i.e. the breakdown period). Based on Equation (14), the probability that a core does not fail over a period of time $t$ can be modeled as:

$$ E_c(t) = e^{-t(\mu)^{\beta}} $$

A core in a shipped chip works properly over the period of time $t$ if it is initially defect-free and it does not fail over a period of time $t$. Therefore, its probability can be expressed as:

$$ F_c(t) = q_c \times e^{-t(\mu)^{\beta}} $$

Where $q_c$ is the probability that a core is defect-free given that it passes the test, which is defined in Equation (4). According to [27], the common range of $\beta$ in the infant mortality period is 0.2 to 0.6.

![Figure 1: A core’s life cycle can be classified into three distinct periods: infant mortality, grace, and breakdown.](image-url)
In a shipped chip with \( m \) active cores and \( s \) spares, up to \( s \) failed active cores can be replaced by spares if and only if the failures can be detected by in-field testing, and the replaced spares have not yet failed. This would lead us to the following model for \( F(s,t) \):

\[
F(s,t) = \sum_{i=0}^{s} \left( \begin{array}{c} m \\ i \end{array} \right) \left( 1 - F_c(t) \right)^{m-i} \left( F_c(t) \right)^{i} \times \theta \end{array} \right) ^{i} E(i,s,t) \tag{17}
\]

\[
E(i,s,t) = \sum_{j=m}^{s} \left( \begin{array}{c} s \\ j \end{array} \right) \left( 1 - F_c(t) \right)^{j} \tag{18}
\]

Where \( \theta \) is the defect-coverage of in-field testing, and \( E(i,s,t) \) is the probability of finding at least \( i \) out of \( s \) spares that have not failed at time \( t \).

### 4.3 Burn-in Cost Model

So far we have modeled the manufacturing cost and service cost without burn-in. In the following, we include burn-in effects in the equations.

Burn-in is an expensive process; so we need to add its cost to the manufacturing cost. In addition, yield after burn-in is less than the yield before burn-in. Considering these two changes, Equation (8) should be revised as follows:

\[
C_{Man}^{b} = \frac{nC_F^{b} + C_{Burn}^{b}}{y_{sys}^{b}} \tag{19}
\]

Where the \( C_{Burn} \) is the cost of burn-in per chip, and \( y_{sys}^{b} \) is the observed system yield after burn-in. Similar to Equation (3) we can express \( y_{sys}^{b} \) in terms of \( y_{c}^{b} \) as follows:

\[
y_{sys}^{b} = \sum_{i=m}^{n} \left( \begin{array}{c} n \\ i \end{array} \right) \left( 1 - y_{c}^{b} \right)^{n-i} \left( y_{c}^{b} \right)^{i} \tag{20}
\]

Where \( y_{c}^{b} \) is the observed core yield after burn-in. The difference between \( y_{c}^{b} \) and \( y_{c}^{s} \) is yield drop due to the new defective cores accelerated by the burn-in process. Therefore, we can model \( y_{c}^{b} \) as:

\[
y_{c}^{b} = y_{c}^{s} \times E_c^{b}(t_0) = \left( 1 + \frac{\lambda A \Omega}{\alpha} \right)^{-\alpha} \times e^{-\left( \frac{t_0}{\mu/\Omega} \right) ^{\beta}} \tag{21}
\]

Here \( t_0 \) is the end of the infant mortality period, as after burn-in the chip should be in the grace period of its lifecycle. Failures continue to occur in this period, and the service cost analysis from Section 4.2 can be applied again with minor changes: the shape parameter \( (\beta) \) is now equal to one, and \( y_{c}^{s} \) and \( y_{sys}^{s} \) in Equations (9)-(18) are replaced with \( y_{c}^{b} \) and \( y_{sys}^{b} \):

\[
C_{Ser}^{b} = (1 - R_{F}^{b})C_{F}^{b} \tag{22}
\]

\[
R_{F}^{b} = \sum_{s=0}^{n-m} P^{b}(s) \times F^{b}(s,t) \tag{23}
\]

\[
P^{b}(s) = \frac{R^{b}(s)}{y_{sys}^{b}} \tag{24}
\]

\[
R^{b}(s) = \left( \begin{array}{c} n \\ m+s \end{array} \right) \left( 1 - y_{c}^{b} \right)^{n-m-s} \left( y_{c}^{b} \right)^{m+s} \tag{25}
\]

\[
F_{c}^{b}(t) = q_{c}^{b} \times e^{-t/\mu} \tag{26}
\]

\[
q_{c}^{b} = \frac{y_{c}^{b}}{y_{c}^{b}} \tag{27}
\]

\[
y_{c}^{b} = y_{c}^{s} \times E_{c}^{b}(t_0) = \left( 1 + \frac{\lambda A \Omega}{\alpha} \right)^{-\alpha} \times e^{-\left( \frac{t_0}{\mu/\Omega} \right) ^{\beta}} \tag{28}
\]

\[
F^{b}(s,t) = \sum_{i=m}^{n} \left( \begin{array}{c} m \\ i \end{array} \right) \left( 1 - F_{c}^{b}(t) \right)^{n-i} \left( F_{c}^{b}(t) \right)^{i} \times \theta \tag{29}
\]

\[
E^{b}(i,s,t) = \sum_{j=m}^{s} \left( \begin{array}{c} s \\ j \end{array} \right) \left( 1 - F_{c}^{b}(t) \right)^{j} \tag{30}
\]

### 5. Experimental Results

In this section, we apply various sets of sample values into our analytical model and illustrate the effect of different factors on the yield, manufacturing cost, and total cost of the system.

Table I shows a list of all input parameters in our analysis along with their corresponding descriptions and typical values. In all of our experiments, we use a 9-out-of-n-core system for illustration. This means that we actually need a multi-core chip with \( m=9 \) active cores, but we put \( s=n-m \) spare cores in the system to replace the possibly failed cores before as well as after shipping. The number of spares considered ranges from zero up to nine.

Based on [7], we set the clustering parameter in our experiments to 2. The scale parameter \( (\mu) \) in Equation (14) is set to 10^6, and the shape parameter \( (\beta) \) is set to 0.4 or 0.6 for the infant mortality period and 1.0 for the grace period. With these parameters, the probability that a chip does not fail within the infant mortality period (i.e. \( E_{c} \)) is 97.52% or 99.6%, and the probability that a chip does not fail within the grace period is 99.99%. In both cases, we set \( t_0=t=10^6 \) hours=1.14 years, assuming that the infant mortality period and the warranty period each lasts about one year.

We normalize the manufacturing cost of each core (i.e. \( C_{K} \)) to one unit, and based on this normalization base, the service cost per failed chip (i.e. \( C_{F} \)) is set to 500. We assume that the burn-in cost is 20% of the manufacturing cost of the chip (i.e. \( C_{Burn} = 0.2 \times C_{K} \)).
Table I: The list of input parameters in our analysis with their descriptions and ranges of values.

<table>
<thead>
<tr>
<th>Param.</th>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n$</td>
<td>Number of cores</td>
<td>9-18</td>
</tr>
<tr>
<td>$m$</td>
<td>Number of active cores</td>
<td>9</td>
</tr>
<tr>
<td>$s$</td>
<td>Number of spares</td>
<td>0-9</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>Clustering parameter</td>
<td>2</td>
</tr>
<tr>
<td>$\Omega$</td>
<td>Defect coverage of manufacturing testing</td>
<td>60%-100%</td>
</tr>
<tr>
<td>$y_c$</td>
<td>True core yield</td>
<td>80%-100%</td>
</tr>
<tr>
<td>$t_0$</td>
<td>Infant mortality period (hour)</td>
<td>10000</td>
</tr>
<tr>
<td>$t$</td>
<td>Warranty period (hour)</td>
<td>10000</td>
</tr>
<tr>
<td>$\mu$</td>
<td>Scale parameter</td>
<td>$10^8$</td>
</tr>
<tr>
<td>$\beta$</td>
<td>Shape parameter</td>
<td>0.4, 0.6, 1</td>
</tr>
<tr>
<td>$\theta$</td>
<td>Defect coverage of in-field testing</td>
<td>60%-100%</td>
</tr>
<tr>
<td>$C_K$</td>
<td>Cost of manufacturing per core</td>
<td>1 (normalized)</td>
</tr>
<tr>
<td>$C_{Burn}$</td>
<td>Burn-in cost per chip</td>
<td>0.2 $n.C_K$</td>
</tr>
<tr>
<td>$C_F$</td>
<td>Service cost per failed chip</td>
<td>500</td>
</tr>
</tbody>
</table>

In Section 5.1, we consider the effect of spares on the system yield, and then we investigate the effect of spares on the manufacturing and total costs in Section 5.2. In Section 5.3, we illustrate the effect of burn-in on the total cost and demonstrate that with a few spare cores in the system, we can actually eliminate the burn-in process and further reduce the cost. Finally, in Section 5.4, we show that when we have spare cores in system, the quality of online testing (or of in-field testing in general) is much more important than that of manufacturing testing.

5.1 System Yield versus Number of Spares

Figure 2 shows the true system yield for different numbers of spares, different core yields, and different defect coverage values. When there are no spare cores, the system yield is low. As we increase the number of spares ($s$), the chance of successful replacement of a failed core by a working spare core increases, and therefore, the system yield increases. After increasing $s$ to a certain amount, the system yield saturates because now the core yield and the defect coverage of the test method become the bottlenecks. Different charts in Figure 2 show that for a lower core yield, we need more spares to compensate for the lost yield.

5.2 Manufacturing and Service Cost

In the previous section, we showed that having spares in the system can significantly improve the system yield. Now, we want to see how this would affect the manufacturing cost. Based on Equation (8), increasing the number of spares would, on the one hand, increase the number of cores and thus the cost. On the other hand, it increases the observed system yield which would reduce the cost. There is an optimal value for $s$, which is a function of the core yield and defect coverage, such that the overall manufacturing cost per shipped chip is minimized. Figure 3 shows the trend of manufacturing cost changes with an increase in the number of spares.

Figure 3 indicates that the manufacturing cost of a shipped chip decreases if a small number of spares are added. After a certain point, the cost starts to increase linearly with respect to the count of added spares. If the true yield of a core is perfect (i.e. $y_c=100\%$), there is no need to have spares because the system yield is already 100% without any spare; therefore, increasing the number of spares results in a linear increase of the normalized manufacturing cost from 9 to 18 (see the bottom-most curve in Figure 3). When the core yield drops further, we need some spares in the system to replace failed cores. The lower the core yield, the more spares are needed to achieve a minimal cost. For example, at $y_c=96\%$, the manufacturing cost would be minimum if one spare is added, while at $y_c=90\%$, two spares would be needed to achieve the minimum manufacturing cost.
So far, we have just considered the manufacturing cost. If we consider the service cost as well, then we need more cores to minimize the total cost. Figure 4 shows the total cost versus the number of spares when the defect coverage, for both manufacturing and in-field testing, is 90%.

In this experiment, we assume that we ship the chips without applying burn-in. Instead, a shipped chip will experience the infant mortality period in the field. Because of this phenomenon, we need to have more spares in the system. Even if the true manufacturing yield of a core is 100%, we need one spare to cover the potential in-field failures for overall service cost minimization (see the bottom curve of Figure 4). As the core yield drops, more spares are needed in the system to minimize the total cost. For example, at $y_c = 90\%$, the total cost would be minimum if five spare cores are included in the chip (see the top-most curve of Figure 4).

**5.3 Burn-in**

So far, we considered the effect of spares in the cost of a chip that is not exposed to a burn-in process. In the following, we assess the benefits of the spare scheme for chips that follow a burn-in step and start their lifecycles from the grace period of the bathtub curve. Furthermore, we evaluate the effects on cost for eliminating the burn-in step and just relying on the spare scheme to repair the in-field failures.

Figure 5 shows the total cost versus the number of spares for a chip that is shipped to the customer after passing manufacturing testing, which includes a burn-in process. Defect coverage for both manufacturing and in-field testing is again fixed at 90% for this experiment. Comparing this figure with Figure 4, we observe that if there is no spare included in the system, burn-in could successfully reduce the cost. In contrast, by comparing the minimum point of each curve in Figure 4 with the minimum point of the corresponding curve in Figure 5, no burn-in achieves a lower minimum cost for each case. We plot the differences between the corresponding curves of Figures 4 and 5 in Figure 6. This shows that with four or more spares in the chip, the burn-in process increases the overall cost. In other words, if we have enough spares in the chip, eliminating the burn-in process is beneficial for overall cost reduction. This conclusion is one of the main contributions of this paper.
5.4 Manufacturing vs. In-field Testing

In our analytical model, we have considered the defect coverage of manufacturing testing ($\Omega$), and the defect coverage of in-field testing such as on-line checking or off-line built-in self-test ($\theta$). In this subsection, we investigate these parameters’ effects on chip cost in the presence of spares. In all the experiments of this section, we assume true core yield is 95%, burn-in is not employed in the manufacturing testing, and $\beta$ is set as 0.4 for the infant mortality period.

In Figure 7, we plot the total cost versus the number of spares for different defect coverages achieved by manufacturing testing. The in-field defect coverage is fixed at 90% in this experiment. If there are no spares in the system, and thus no in-field recovery, the defect coverage of manufacturing testing is a critical factor in determining the total cost. With some spares added into the chip, the curves for different defect coverages are relatively close to each other. These results indicate that with an in-field recovery capability using spares, the manufacturing defect coverage’s criticality to overall cost is significantly reduced. In other words, we can employ a simpler, faster, and cheaper solution to manufacturing testing to reduce the cost.

Figure 7: Total cost versus the number of spares for different manufacturing defect coverage values.

For the next experiment, we fix the defect coverage of manufacturing testing to 90%, and play with the in-field defect coverage to evaluate its effect on the total cost. Figure 8 shows that, with some spares included in the chip, the in-field defect coverage is a very significant factor for the total cost. For example, with four spares, the chip cost with $\theta=60\%$ is 64.36, which is 23% greater than that of the same chip with $\theta=70\%$, and is 4.7 times that of the case in which $\theta=100\%$.

From these experiments we conclude that in a spare-enhanced chip, the in-field test quality is much more important than the manufacturing test quality. An in-field test solution, either on-line checking or BIST, with high defect coverage is necessary.

6. Conclusion

In this paper, we present an analytical model for the yield and cost of a multi-core chip that takes into account the following factors: (1) the number of cores, (2) the number of spares, (3) the core yield, (4) the manufacturing test quality, (5) the in-field test quality, and (6) the burn-in process. Based on this model, we experimentally investigated the effect of each of these factors on the total cost of the chip.

The main contributions of this paper are as follows:

1. Providing an analytical model for the cost of a spare-enhanced multi-core system that has a manufacturing test process with or without burn-in.
2. Showing that adding a few dynamically reconfigurable spares in the chip can significantly reduce the overall cost.
3. Investigating the possibility of removing the burn-in process and relying on the available spares to replace the cores that fail within the infant mortality period.
4. Illustrating that the in-field test quality is much more important than the manufacturing test quality for spare-enhanced chips.

7. Acknowledgement

The authors acknowledge the support of the Gigascale Systems Research Center (GSRC), one of five research centers funded under the Focus Center Research Program, a Semiconductor Research Corporation program.

8. References


