NoC Interconnect Yield Improvement Using Crosspoint Redundancy

Cristian Grecu[1], André Ivanov[1], Res Saleh[1], Partha Pratim Pande[2]

[1] SoC Research Laboratory
Department of Electrical and Computer Engineering
University of British Columbia
2332 Main Mall Vancouver, BC, V6T 1Z4 Canada
{grecuc, ivanov, res}@ece.ubc.ca

[2]School of Electrical Engineering & Computer Science
Washington State University PO Box 642752
Pullman, WA 99164-2752, USA
pande@eecs.wsu.edu

Abstract

Systems-on-chip integrate increasingly larger numbers of pre-designed cores interconnected through complex communication fabrics. For nanometer-scale VLSI processes (45 nm and below), it is difficult to guarantee correct fabrication with an acceptable yield without employing design techniques that take into account the intrinsic existence of manufacturing defects. In order to improve the yield and reliability of multi-core SoCs, their interconnect infrastructures must be designed such that fabrication and life-time faults can be tolerated. In this work we present a self-repair method for the interconnect fabrics of integrated multi-core systems. Our method is based on the use of redundant links and crosspoints, and improves both post-manufacturing yield and life-time reliability of on-chip communication fabrics. Our method can provide a significant interconnect yield improvement (up to 72% in our experiments), and allows fine-tuning of yield versus redundant components.

1. Introduction

A prominent research direction in the design of large integrated systems is the network-on-chip (NoC) paradigm [1], which has emerged as a solution capable of overcoming issues regarding performance, energy consumption, and reusability. The main driver behind the NoC approach is the advance in manufacturing technologies, which allows the integration of hundreds of processing/storage cores on a single chip. These cores are connected in a plug-and-play fashion by an infrastructure consisting of intelligent switches and global interconnects. The global interconnects (inter-switch links) are laid out on upper metal layers at latter stages of the NoC design flow, while the local interconnects are on the lower metal layers.

A major challenge that SoC design [1] is expected to face is related to the intrinsic unreliability of the communication infrastructure under technology limitations. Two different facets can be identified: the first one has its roots in the increasing levels of defect densities that are associated with the continuous feature size shrinking in semiconductor fabrication technology. Chips are more and more prone to defects like shorts and opens caused by particles, contamination, cracks, scratches, and missing vias; photolithography with sub-wavelength feature sizes can result in feature corruption, and chemical mechanical polishing (CMP) may cause surface dishing through over-polishing, ultimately becoming important yield-loss factors [2]. The second aspect is related to the different types of reliability faults that may manifest during the life cycle of the semiconductor chips. The nature of these faults can be permanent (when they reflect an irreversible physical change), intermittent (activated by environmental factors such as voltage or temperature - often becoming permanent faults over time) and transient (caused by neutron and alpha particles, power supply noise, electrostatic discharge and electromagnetic interference) [3].

Proceedings of the 21st IEEE International Symposium on Defect and Fault-Tolerance in VLSI Systems (DFT'06)
0-7695-2706-X/06 $20.00 © 2006 IEEE

IEEE COMPUTER SOCIETY

Authorized licensed use limited to: Washington State University. Downloaded on June 17,2010 at 16:31:40 UTC from IEEE Xplore. Restrictions apply.

To overcome the abovementioned problems in the particular case of the on-chip communication infrastructures, we propose a design methodology where the NoC interconnect incorporates yield-enhancing features and defect tolerant techniques. The structured nature and simple functionality of the NoC interconnect infrastructures render them particularly suitable for the implementation of these features (similar to the case of memory blocks, where yield enhancement and defect tolerant techniques have become common place [4]). The impact of these techniques on the overall yield of a NoC-based chip can be quite significant, given that the data communication infrastructure alone can consume around 10-15% of the total chip area [5]. More importantly than the area argument, the NoC inter-switch links are likely to be placed on the upper metal layers, and hence, the layout scenario is very similar to the situation shown in Fig. 1, with the active devices (cores, switches) on the silicon surface, and the inter-switch wires on the higher levels of the 3D structure.

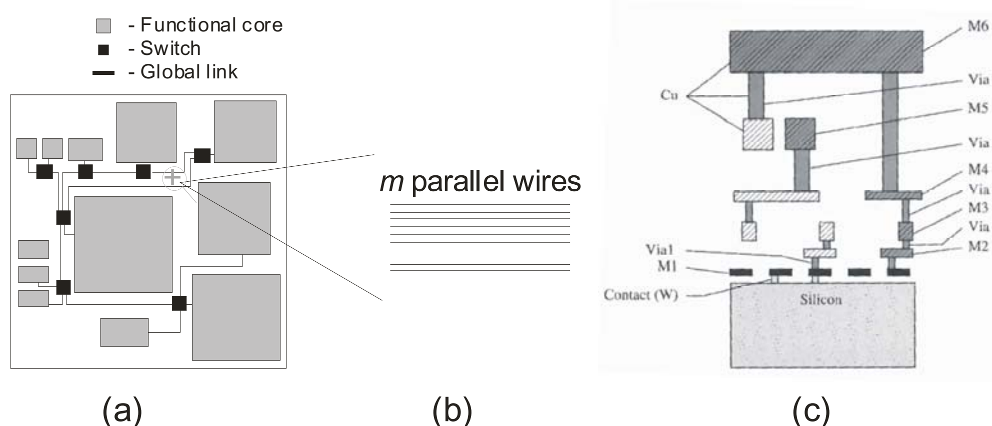

Fig. 1: (a) Global links in NoC-based systems-on-chip; (b) global interconnect segment with *m* signal lines; (c) global line spanning multiple metal/via levels.

The global signal lines are more prone to manufacturing defects, since they span more metal layers and require more vias in order to reach the active devices on the silicon surface. The increase in number of metal layers is driven by the higher integration level of semiconductor devices and a direct consequence of Moore's law. Hence, it is expected that the yield of global interconnects will decrease as the number of metal layers increases, as projected by the ITRS documents [6] (e.g., 12 metal layers for 45 nm technology, compared with 9 layers for a typical 130 nm process). In this work, we propose a method to improve the yield of global interconnects of NoC-based chips by using redundant upper-level wires interconnected with the low-level metal lines in a sparse matrix configuration. This method is based on the use of sparse regular crossbars, which are a particular class of generalized connectors.

2. Related work

Recently, the multi-core, multi-processor SoCs have been proposed and designed to deliver the computational power for demanding multimedia and communications applications. The typical number of cores in such SoCs is in the range of 100 or more, and their internal data throughput can reach up to 100 Gbps [7].

Sparse crossbars were proposed originally in [8] for designing high-performance, hardware efficient concentrators and distributors for interconnection networks. Their applications range from computer networks [9] to multi-processor systems [10] and instrumentation networks [11].

Fault tolerance of global SoC interconnects was addressed by different groups proposing error recovery schemes based on error detection and correction [12], mainly aimed at

IEEE COMPUTER SOCIETY
Authorized licensed use limited to: Washington State University. Downloaded on June 17,2010 at 16:31:40 UTC from IEEE Xplore. Restrictions apply.

recovering from failures caused by transient faults. For more structured interconnect systems, such as FPGAs, tolerance to permanent faults can be achieved by rerouting and remapping the faulty interconnect resources [13]. The advantage of the FPGA systems from this point of view is their high level of programmability and modularity, which make them readily adaptable for fault tolerance.

The NoC global interconnects (inter-switch links) do not have the degree of programmability of FPGA interconnects; however, they are prone to the same types of manufacturing and reliability defects. Although most of these defects can be detected during manufacturing test, permanent defects (undetected and/or developed post-fabrication) may still affect the operation of SoC interconnects due to latent manufacturing flaws and wear-out mechanisms. More importantly, the global interconnect defects are a potential source of yield loss [6] due to the large number of vias they need.

In this paper, we propose a method to design global NoC interconnects such that a specified interconnect yield target can be achieved. We present the trade-off between the interconnect resources and the achievable yield.

3. Problem statement: assumptions and requirements

In this section we formally describe the problem of constructing global NoC interconnects such that, under a specific set of defect assumptions, a certain interconnect yield be achieved. Let L be an interconnect segment consisting of a set of wires whose function is to transfer a certain number of signals, denoted by m. Let P_{line} be the probability that a single global line (including the vias from the silicon surface to the upper metal layers and back – refer to Fig. 1) is fabricated correctly. Given a certain interconnect yield target Y to be achieved, we want to determine the total number of metal lines n ($n \geq m$) that need to be provided to the interconnect segment L.

Additional constraints are:

1. The spare and active metal lines must have closely matched electric characteristics and must be *fully interchangeable*.
2. The switching circuitry that routes the m signal lines to the n active wires of the interconnect L must be of *minimal area and complexity*.

The first requirement above reflects the fact that the circuits that implement and manage the interconnect and perform the actual mapping of signal lines to physical wires may place an additional load on the metal wires. If this load is not identical for all wires, than their delays may be different, and, consequently, the timing characteristics of the same interconnect segment may change when the interconnect is reconfigured upon the occurrence of a permanent fault.

The second requirement refers to the fact that the implementation of the fault tolerant features must be hardware efficient, since the silicon area is an important component of the overall cost.

Our approach is intended for use in the physical design stage of the chip-making process and is aimed specifically at NoC global links. It consists of the following steps:

Step 1: Identify the global signal lines that connect different cores and span multiple metal layers (as indicated by the example in Fig. 1).

Step 2: Calculate the probability of failure per line, using the probability of failure of an individual via; calculate the number of spare metal lines required in order to achieve the desired interconnect yield.

Step 3: Provide a mechanism that can select a set of good lines at both ends of the interconnect segment, with the number of defect-free lines equal to the number of logic core I/Os, from a total number of lines equal to the sum of logic I/Os and spare lines calculated at *Step 2*.

IEEE COMPUTER SOCIETY

Authorized licensed use limited to: Washington State University. Downloaded on June 17,2010 at 16:31:40 UTC from IEEE Xplore. Restrictions apply.

In the following section, we present a method to calculate the number of spare metal lines that must be provided in order to achieve a desired interconnect yield.

4. Interconnect yield modeling and spare calculation

In VLSI manufacturing, the critical area model [14] is commonly employed to estimate the yield sensitivity of chips to random failure mechanisms. For specific manufacturing processes, the results are then calibrated using electrical test structures [18]. We apply the power law model, following the procedure in [15], and the critical area model to determine the probability of failure of an individual via, $PoF_{via:}$.

$$PoF_{via} = D_{0,ref} \cdot Dens^{k-1} / Dens_{ref}^{k} \qquad (1)$$

Here, $D_{0,ref}$ and $Dens_{ref}$ are constants for a specific manufacturing process and determined through curve fitting during process characterization. It has been observed [15], through experiments on technologies with different minimum feature sizes, that the exponent of the power law model, k, tends to 1 as the feature size decreases. This allows us to make the assumption that PoF_{via} is (almost) independent of via density (*Dens*) and consider via failures as independent events for statistical yield calculations. This weak dependence is more pronounced for smaller feature sizes, i.e., PoF_{via} in a 90 nm technology has a weaker dependence on via density than PoF_{via} in a 130 nm technology.

We then determine the probability of failure of an interconnect line spanning multiple metal layers P_{line} taking into account the corresponding number of via levels that it traverses. Assuming via failures on different via levels as independent events, the probability that a global interconnect line spanning l metal layers is defect-free can be expressed as:

$$P_{line} = \prod_{i=1}^{l} \left(1 - PoF\left(via_{l-1,l}\right)\right)^{2} \qquad (2)$$

Under these considerations, tolerating global wire defects is a matter of providing adequate number of spares, separating the good from the bad, and configuring the inter-core links accordingly.

We can perform an m-choose-n calculation to obtain the number of wires that must be available (n) to achieve a given number of functional wires (m) in the link. The probability that the yield be such that exactly i wires are defect-free is:

$$P_{yield}(n,i) = \left(\binom{n}{i} \left(P_{line}\right)^{i} \left(1 - P_{line}\right)^{n-i} \right) \qquad (3)$$

That is, there are exactly $\binom{n}{i}$ ways of selecting i wires from a total of n wires, and the probability that each of these cases is defect-free is $\left(P_{line}\right)^{i}\left(1-P_{line}\right)^{n-i}$. We can have a set of m functional wires whenever m or more of them are defect-free, so the yield of the link is expressed as the cumulative distribution function:

IEEE COMPUTER SOCIETY

Authorized licensed use limited to: Washington State University. Downloaded on June 17,2010 at 16:31:40 UTC from IEEE Xplore. Restrictions apply.

$$P_{m\ of\ n} = \sum_{i=m}^{n} \left(\binom{n}{i} (P_{line})^{i} (1 - P_{line})^{n-i} \right) \qquad (4)$$

Given the desired probability (or yield) for having at least m defect-free wires, $P_{m\ of\ n}$, Eq. (4) provides a way to find the number of physical wires, n, that are required in the layout in order to achieve the target yield.

5. Link configuration mechanism

In order to physically connect a set of m functional wires at the core interface, we must provide a way to arbitrarily connect the m interface signals to the n physical wires ($m \leq n$) such that exactly m defect-free wires are used to transmit the data between two cores. Moreover, we must ensure that in any possible configuration, the electrical properties of the core-to-core interconnect are preserved, i.e., the delays of different possible configurations must match closely. Another constraint that we have is that the area of the configuration circuitry must be minimal, since this adds up to the total cost of interconnects.

With these restrictions, we propose a method to construct configurable connectors that have identical delays for any input/output connection, and are of minimal size. This method is based on the theory of generalized connectors [8], and uses specific results for a class of connectors known as regular sparse crossbar concentrators [16].

5.1. Sparse crossbars

An (m,n)-sparse crossbar is defined as a bipartite graph $G=\{I,O,E\}$, with a set of m inputs $\{I\}$, a set of m outputs $\{O\}$, and a set of edges $\{E\}$ such that there exist a one-to-one mapping between any n or fewer outputs and the m inputs. The edges in E are called the crosspoints of graph G. The number of crosspoints of G represents its crosspoint complexity. For a hardware implementation of a sparse crossbar, the crosspoint complexity is a direct measure of the silicon area required by its layout.

The number of outputs (inputs) to which an input (output) in connected to is called its *fanout* (*fanin*), and the maximum number of outputs (inputs) to which an input (output) in G is connected is called the *fanout* (*fanin*) of G. A sparse crossbar is said to be *regular* if the difference between both the *fanouts* of its inputs and the *fanins* of its outputs is no greater than two.

The direct sum $G_1 + G_2$ of sparse crossbars $G_1=\{I,O_1,E_1\}$ and $G_2=\{I,O_2,E_2\}$ is another sparse crossbar $G_1+G_2=\{I,O_1UO_2,E_1UE_2\}$.

An (m,n)-sparse crossbar is called a *fat-and-slim* crossbar if any n-m of its outputs are connected to all the inputs in I, and each of the remaining n outputs is connected to a distinct input. We can represent an (m,n) fat-and-slim crossbar G by an m x n adjacency matrix, $A_G=[a_{ij}]_{m\ x\ n}$ of binary elements, where a '1' element in row i and column j represents a crosspoint between input i and output j. An (m,n)-fat-and-slim crossbar has the adjacency matrix $[F_{m,\ n\text{-}m}|S_m]$, where $F_{m,\ n\text{-}m}$ is a m x n-m matrix of '1's and S_m denotes the m x m identity matrix, as indicated in the example in Fig. 2.

IEEE COMPUTER SOCIETY

Authorized licensed use limited to: Washington State University. Downloaded on June 17,2010 at 16:31:40 UTC from IEEE Xplore. Restrictions apply.

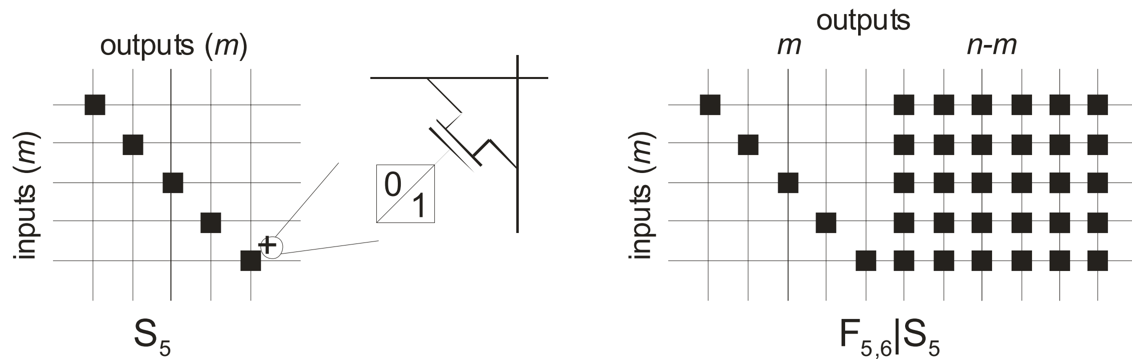

Fig. 2: (a) Non-fault-tolerant sparse crossbar and crosspoint implementation; (b) *n-m* fault-tolerant imbalanced sparse crossbar.

5.2. Construction of fault tolerant global interconnects based on regular crossbars

Based on the yield considerations formulated in Sec. 4 and sparse crossbars briefly described in Sec. 5.1, we now present a method to build connectors that can tolerate defects of the global wires, while still complying with our delay and area restrictions.

As shown in Fig. 2, we start with a simple crossbar whose corresponding adjacency matrix is the *m x m* identity matrix. This corresponds to the case when the global interconnect is not fault tolerant and represents the 'slim' part of the fat-and-slim crossbar. We then add the 'fat' part corresponding to *n-m* spare wires required to achieve the desired interconnect yield according to Eq. (4), a full crossbar whose adjacency matrix is of size *(n-m) x (n-m)* and has all elements'1'.

We have now a fat-and-slim crossbar that can tolerate *n-m* defects. This crossbar is not balanced, in the sense that the capacitive load from an input to an output is not matched for different input-output combinations. To balance the load, we can transform the fat-and-slim crossbar such that for each input-output combination, the sum of the corresponding fanout and fanin is matched closely.

First, we need a proof that such a crossbar exists. The proof is provided by the following theorem [16]:

Theorem 1: For any positive integers *m* and $n \geq m$, the *(m,n)*-fat-and-slim sparse crossbar can be rearranged to obtain a sparse crossbar with fanout of either $\alpha \pm 1$ or α, fanin of $n-m+1$ and minimum number of crosspoints, where α is given by:

$$\alpha = \begin{cases} \left\lfloor \frac{(n-m+1)m}{n} \right\rfloor, & \text{if } n < \frac{3m}{2} \\ \left\lfloor \frac{(n-m+1)m}{n} \right\rfloor, & \text{if } \frac{(n-m+1)m}{n} - \left\lfloor \frac{(n-m+1)m}{n} \right\rfloor < 0.5, \text{ and } n \geq \frac{3m}{2} \\ \left\lceil \frac{(n-m+1)m}{n} \right\rceil, & \text{if } \frac{(n-m+1)m}{n} - \left\lfloor \frac{(n-m+1)m}{n} \right\rfloor \geq 0.5, \text{ and } n \geq \frac{3m}{2} \end{cases} \quad (5)$$

where $\lfloor x \rfloor$ and $\lceil x \rceil$ are the floor and ceiling functions, defined as the largest integer less than or equal to x, and the smallest integer not less than x, respectively. For a rigorous mathematical proof of this theorem, the reader is referred to [16] and [17].

Intuitively, we can observe that the sums *fanin+fanout* of input-output pairs can differ by one due to the *fanin* and *fanout* being even or odd integers, which explains why the capacitive loads of input-output connections, cannot always be made identical; electrically, this implies

0-7695-2706-X/06 $20.00 © 2006 IEEE
IEEE COMPUTER SOCIETY
Authorized licensed use limited to: Washington State University. Downloaded on June 17,2010 at 16:31:40 UTC from IEEE Xplore. Restrictions apply.

that they may nominally differ by an amount equal to the equivalent nominal capacitance of one crosspoint. Once we know that such crossbars exist, we need a way to build them.

We use the following column exchange operation to balance the rows and columns of the adjacency matrix of a fat-and-slim crossbar. Let A_G be the adjacency matrix of the (m,n)-sparse crossbar G. Let x and y be any two columns in A_G, where x covers y. Let $a_{i1,x}, a_{i2x}, \ldots, a_{ir,x}$ be a string of r '1's in x. For any $1 \leq l \leq r$, the column exchange operation is performed by exchanging $a_{il,x}$ with $a_{il,y}$. If matrix B is the result of the column exchange operation, then, when no column x is left that covers any other column y, matrix B is balanced. Fig. 3 shows how the column exchange operation is applied on the fat-and-slim, *n-m* fault tolerant crossbar. We observe that column 6 is the first that covers column 1, so we start by exchanging elements (2,1) and (2,6). For the matrix newly formed, we observe that the first column covered by column 6 is now column 3, so we proceed to exchange elements (4,3) and (4,6). The column exchange operations are continued in this fashion until no column is left that covers another column.

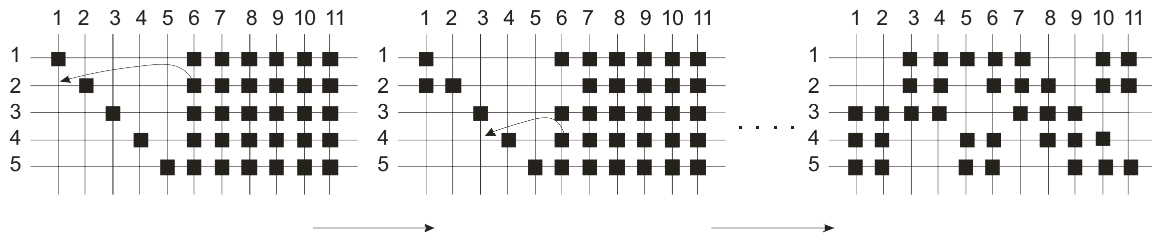

Fig. 3: Balanced, (n-m) fault-tolerant sparse crossbar, after successive column exchange operations

The column exchange operation distributes evenly the crosspoints along rows and columns of the sparse crossbar. For a rigorous proof of this property of the above operation, the reader is referred to [16].

Fig. 3 shows the status of the defect-tolerant sparse crossbar in Fig. 2 after the balancing was done. Note that, for each possible input-output combination, the sums *fanout* + *fanin* do not differ with more than 1.

6. Experimental results and discussion

We applied the method developed in this work to sets of global interconnects of different widths, representative for the cases of network-on-chip communication fabrics. We considered the case of a 65 nm technology with 11 metal layers and defect distributions as projected by the ITRS 2005 roadmaps.

Fig. 4 plots the total number of wires (including the redundant ones), n, that need to be routed in order to provide 90%, 99%, and 99.9% interconnect yield respectively, for global link widths (m) of 32, 64, and 128 bits and probability of defect-free global interconnect line ranging between 0.9 and 1.

IEEE COMPUTER SOCIETY

Authorized licensed use limited to: Washington State University. Downloaded on June 17,2010 at 16:31:40 UTC from IEEE Xplore. Restrictions apply.

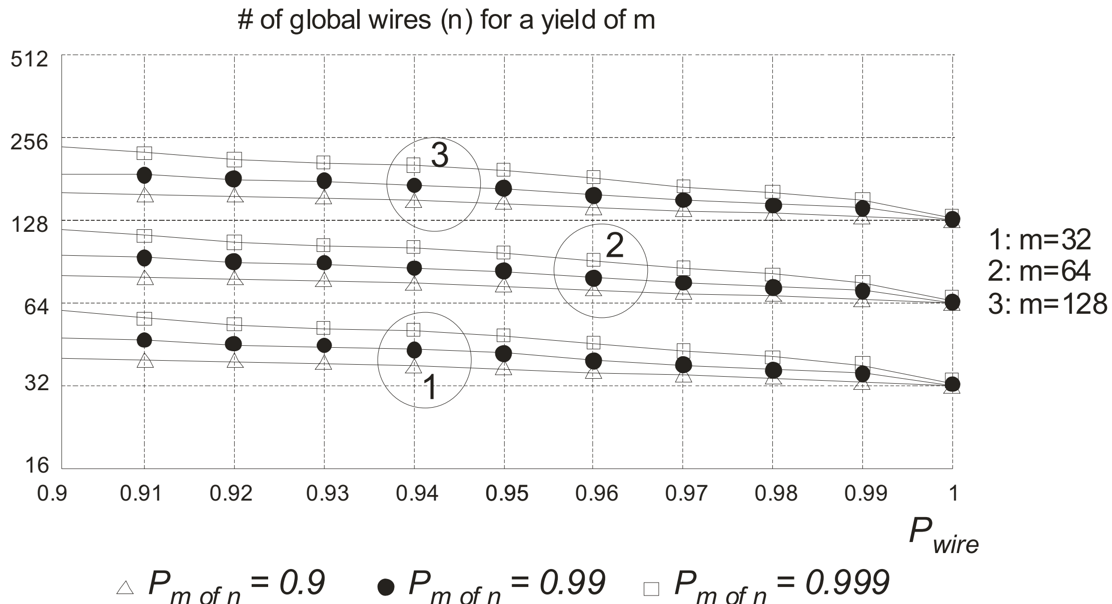

Fig. 4: Number of total lines n for different link widths (32, 64, 128 bits) and target yields ($P_{m\ of\ n}$)

The total number of wires *n,* in the above example, was determined assuming the global interconnects are laid out on upper metal layers. As mentioned above, we use a 65 nm technology, with a defect density of 0.025defects/cm^2 obtained from [6].

Table 1 shows the relative interconnect yield improvement (fourth column in the table) for a probability of defect-free line P_{line} = 0.99 and the cost to achieve it in terms of number of crossbar crosspoints (third column) for different target interconnect yields and link widths.

Table 1: Yield improvement vs. interconnect complexity

Interconnect width (m)	Target interconnect yield $P_{m\ of\ n}$ [%]	Number of crosspoints	Interconnect yield improvement [%]
32	90	96	17.5
	99	288	26.5
	99.9	708	27.4
64	90	260	37.5
	99	548	46.5
	99.9	2180	47.4
128	90	1152	62.5
	99	5312	71.5
	99.9	10128	72.4

As shown in Table 1, the yield of global interconnects can be significantly improved by using our technique. It is, however, extremely important to estimate carefully the target yield and characterize the manufacturing process accurately, since demanding a high yield or underestimating the individual *P(line)* for global interconnects can increase dramatically the size of the fault tolerant crossbars and, implicitly, the silicon area and metal lines required. For example, if we target a 99% yield for a 64-bit wide global link, we need to provide a 548-crosspoint sparse crossbar at each end of the link; if, instead, we aim for a 99.9% yield, the number of required crosspoints is 2180. That translates to an approximately 4X increase in silicon area for a yield improvement of only 0.9%.

Authorized licensed use limited to: Washington State University. Downloaded on June 17,2010 at 16:31:40 UTC from IEEE Xplore. Restrictions apply.

7. Conclusions

We presented a structured method to build defect tolerant global interconnects that takes into account the manufacturing process characteristics in terms of via failure probability and, in turn, generates interconnects with the desired target yield. The mechanism that selects and reconfigures the interconnect is based on balanced sparse crossbars, and uses a minimum number of crosspoints to connect the set of defect-free wires at the core interface. The method also ensures that any valid interconnect configuration has identical capacitive load, which guarantees the invariance of timing characteristics of different possible configurations. An important advantage of this technique is that it can be easily automated and implemented in SoC design tools.

We intend to enhance the method by considering additional interconnect failure mechanisms, such as opens and shorts. We are also in the process of developing a complete autonomous repair solution for interconnects, including self-test and configuration hardware, combined with fault tolerant NoC switches We expect the solution we proposed to become more relevant with the emergence of nanometer-scale VLSI processes, where high defect rates and device failures will be serious manufacturing limitations.

8. References

[1] P. Magarshack, P. G. Paulin, "System-on-chip beyond the nanometer wall," *Proceedings of the 40th Design Automation Conference (DAC'03)*, 2003, pp:419-424.

[2] Y. Zorian, D. Gizopoulos, C. Vandenberg, P. Magarshack, “Guest editors' introduction: design for yield and reliability”, *IEEE Design & Test of Computers*, May-June 2004, Vol. 21, Issue: 3, pp: 177–182.

[3] C. Constantinescu, “Trends and challenges in VLSI circuit reliability*”, IEEE Micro*, July-Aug. 2003, Vol. 23, Issue: 4, pp: 14-19.

[4] S. Shoukourian, V. Vardanian, Y. Zorian, “SoC yield optimization via an embedded-memory test and repair infrastructure”, *IEEE Design & Test of Computers*, May-June 2004, Vol. 21 , Issue: 3, pp: 200 – 207.

[5] P. P. Pande, C. Grecu, M. Jones, A. Ivanov, R. Saleh, “Performance evaluation and design trade-offs for network-on-chip interconnect architectures”, *IEEE Transactions on Computers*, Vol. 54, Issue: 8, Aug. 2005, pp: 1025–1040.

[6] International Technology Roadmap for Semiconductors, 2005 update, http://www.itrs.net/Common/2005ITRS/Home2005.htm

[7] K. Goossens, J. Dielissen, A. Radulescu, “The Aethereal network on chip: concepts, architectures, and implementations”, *IEEE Design and Test of Computers*, Sept-Oct 2005, Vol. 22, Issue: 5, pp:21-1

[8] G. Masson, “Binomial switching networks for concentration and distribution”, *IEEE Transactions on Communications*, Sep 1977, Vol. 25, Issue: 9, pp: 873–883.

[9] E. Dekker, “Architecture scalability of parallel vector computers with a shared memory”, *IEEE Transactions on Computers*, May 1998, Vol. 47, Issue: 5, pp: 614-624.

[10] B. Wilkinson, “On crossbar switch and multiple bus interconnection networks with overlapping connectivity”, *IEEE Transactions on Computers*, June 1992, Vol: 41, Issue: 6, pp:738–746.

[11] K.D. Pedrotti, S. Beccue, W.J. Haber, B.P. Brar, G. Robinson, M.K. Kilcoyne, ”16×16 optical-fiber crossbar switch operating at 0.85 μm”, *Journal of Lightwave Technology*, Sep 1990 ,Vol. 8, Issue: 9, pp:1334-1342.

[12] S. Murali, T. Theocharides, N. Vijaykrishnan, M.J. Irwin, L. Benini, G. De Micheli, “Analysis of error recovery schemes for networks on chips”, *IEEE Design & Test of Computers*, Sept.-Oct. 2005, Vol. 22, Issue: 5, pp: 434-442.

[13] J. Huang; M.B. Tahoori, F. Lombardi, “Routability and fault tolerance of FPGA interconnect architectures”, *Proceedings of the IEEE International Test Conference*, ITC 2004, pp: 479-488.

[14] S. Gandemer, B.C. Tremintin, J.-J. Charlot, “Critical area and critical levels calculation in IC yield modeling*”, IEEE Transactions on Electron Devices*, Feb. 1988, Vol. 35, Issue: 2, pp: 158-166.

[15] D.K. de Vries, P.L.C. Simon, “Calibration of open interconnect yield models”, *Proceedings of the 18th IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems*, 2003, pp:26-33.

[16] W. Guo, A.Y. Oruc, “Regular sparse crossbar concentrators”, *IEEE Transactions on Computers*, March 1998, Vol. 47, Issue: 3, pp: 363-368.

[17] A. Y. Oruc, H.M. Huang, “Crosspoint complexity of sparse crossbar concentrators”, *IEEE Transactions on Information Theory*, Sept. 1996, Vol. 42, Issue: 5, pp:1466-1471.

[18] A. Cabrini, D. Cantarelli, P. Cappelletti, R. Casiraghi, A. Maurelli, Marco Pasotti, P.L. Rolandi, G. Torelli, “A test structure for contact and via failure analysis in deep-submicrometer CMOS technologies”, *IEEE Transactions on Semiconductor Manufacturing*, Feb. 2006, Vol. 19, Issue: 1, pp: 57-66.

Proceedings of the 21st IEEE International Symposium on Defect and Fault-Tolerance in VLSI Systems (DFT'06)
0-7695-2706-X/06 $20.00 © 2006 IEEE

IEEE COMPUTER SOCIETY

Authorized licensed use limited to: Washington State University. Downloaded on June 17,2010 at 16:31:40 UTC from IEEE Xplore. Restrictions apply.