HOKKAIDO UNIVERSITY

Title	Novel Fuse Scheme with a Short Repair Time to Maximize Good Chips per Wafer in Advanced SoCs
Author(s)	Matsumoto, Chizu; Hamamura, Yuichi; Nakao, Michinobu et al.
Citation	IEICE Transactions on Electronics, E96.C(1), 108-114 https://doi.org/10.1587/transele.E96.C.108
Issue Date	2013-01-01
Doc URL	https://hdl.handle.net/2115/52057
Rights	Copyright © 2013 The Institute of Electronics, Information and Communication Engineers
Type	journal article
File Information	ToE96C-1_108-114.pdf

Hokkaido University Collection of Scholarly and Academic Papers : HUSCAP

IEICE TRANS. ELECTRON., VOL.E96–C, NO.1 JANUARY 2013

PAPER

Novel Fuse Scheme with a Short Repair Time to Maximize Good Chips per Wafer in Advanced SoCs

Chizu MATSUMOTO[†a)], *Member*, Yuichi HAMAMURA[†], *Nonmember*, Michinobu NAKAO[††], *Member*, Kaname YAMASAKI[††], Yoshikazu SAITO[††], *Nonmembers, and* Shun'ichi KANEKO[†††], *Member*

SUMMARY Repairing embedded memories (e-memories) on an advanced system-on-chip (SoC) product is a key technique used to improve product yield. However, increasing the die area of SoC products equipped with various types of e-memories on the die is an issue. A fuse scheme can be used to resolve this issue. However, several fuse schemes that have been proposed to decrease the die area result in an increased repair time. Therefore, in this paper, we propose a novel fuse scheme that decreases both die area and repair time. Moreover, our approach is applied to a 65 nm SoC product. The results indicate that the proposed fuse scheme effectively decreases the die area and repair time of advanced SoC products.

key words: *random access memory, system-on-chip, redundancy, fuse*

1. Introduction

Recent mobile applications, such as mobile phones and media tablets, equipped with the system-on-chip (SoC) technology process large volumes of information. Therefore, it is required to package a larger number of embedded memories (e-memories) onto a single SoC die. However, because e-memories require fine processing technologies, the possibility of defects occurring in these e-memories is higher than in other portions of a die. In other words, e-memories are a major yield loss factor.

To improve product yield, we must not only reduce defects in the manufacturing line but also improve manufacturability during the design phase [1]. As technology migration proceeds, the latter becomes more important. From the viewpoint of advanced SoC product design, the effective design of memory redundancy is the key to increasing product yield. Although memory redundancy increases product yield, the number of dies on a wafer decreases owing to increase in die size. Therefore, an effective repair technique that maximizes the number of good dies on a wafer is required.

Numerous approaches have been proposed to maximize the number of good chips on a wafer [2]–[11]. One approach is to optimize the number and types of spare bits (e.g., row, column, and/or IO) of each memory according to manufacturing defects [2]–[4]. Another approach is to decrease the built-in self-test (BIST) and built-in self-repair (BISR) area by shearing the BIST and BISR circuits among some e-memories [5]–[9]. Fuse area reduction is another effective approach [10], [11]. For example, compressing technique of repair information among e-memories results in a reduced fuse area [10]. Moreover, in the works of [11], the fuse area can be reduced by shearing a fuse among e-memories. However, as the number of e-memories on a die increases, cycles required for fuse data transfer increase owning to the serial connection of the fuse to e-memories. This results in longer repair times. To improve the productivity of advanced SoC products with dies that contain a large number of e-memories, fuse schemes are required to be both area efficient and time efficient.

In this paper, we propose a novel fuse scheme that decreases both fuse area and repair time on a die. In the proposed fuse scheme, e-memories are organized into "virtual e-memory groups," and all the e-memories in one virtual e-memory group connect to the same fuse in parallel. When a failure bit is detected in an e-memory, the bit address of that e-memory and the corresponding bit addresses of the other e-memories in the same virtual e-memory group are simultaneously switched to spare addresses. Furthermore, because several virtual e-memory groups are connected to different fuses, fuses of virtual e-memory groups operate individually and in parallel. In the proposed fuse scheme, only several fuses are embedded on a die and only one cycle is required to transfer fuse data. Thereby it is achieved to reduce of both fuse area and fuse data transfer time. In the proposed fuse scheme, because repair rate is affected by the manner in which e-memories are grouped, we present a grouping method to form virtual e-memory groups.

2. Conventional Fuse Schemes and Problems

A fuse switches a failure bit to a spare bit in a row, column, or IO unit. In early years, dies contained only a few fuses connecting to individual e-memories [12]–[14]. Hence, failure bits are independently repaired in different e-memories. In recent years, advanced SoC products contain a large number of e-memories scattered on a die, thus resulting in an increase in the fuse area; this has become a major problem. The fuse area must be decreased because several fuses are actually used during manufacturing. Therefore, several studies have proposed effective fuse schemes [10], [11]. Figure 1 shows the simplified block diagram of a representative

Manuscript received July 23, 2012.

[†]The authors are with the Yokohama Research Laboratory, Hitachi, Ltd., Yokohama-shi, 244-0817 Japan.

[††]The authors are with Renesas Electronics Corp., Kodaira-shi, 187-8588 Japan.

[†††]The author is with Hokkaido University, Sapporo-shi, 060-0814 Japan.

a) E-mail: chizu.matsumoto.rz@hitachi.com

DOI: 10.1587/transele.E96.C.108

Copyright © 2013 The Institute of Electronics, Information and Communication Engineers

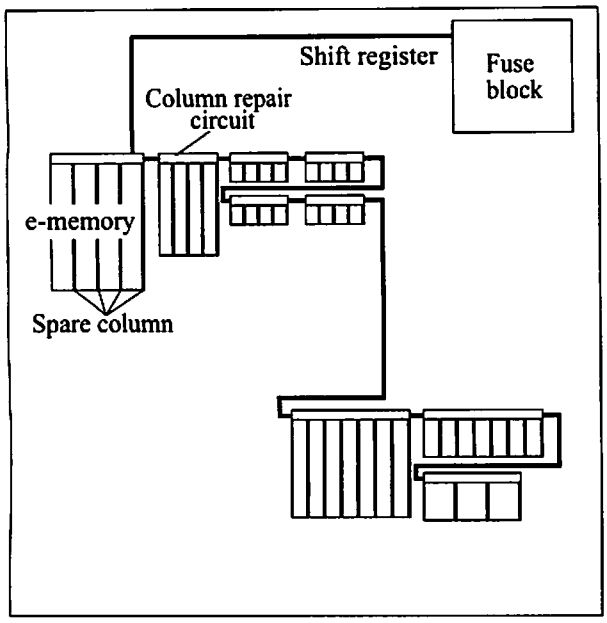


Fig. 1 Block diagram of conventional fuse schemes.

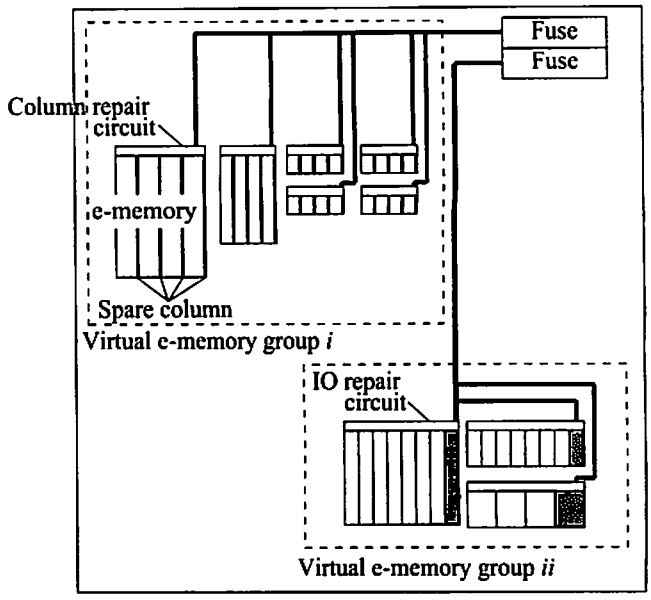


Fig. 2 Block diagram of the proposed fuse scheme.

fuse scheme proposed in [10], [11]. As shown in Fig. 1, a fuse block connects to many e-memories in serial through a shift register. The fuse block consist of several fuses (less than the number of e-memories on a die) and switches several failure bits to the corresponding spare bits in the e-memories. For example, when a failure bit is detected in an e-memory, a fuse switches the failure bit to a spare bit, and the failure bits are sequentially repaired through the shift resister chain. Since only several bits cause failures on a die, sharing several fuses among many e-memories is an effective technique to decrease the fuse area while maintaining a high repair rate. However, this fuse scheme has a drawback with respect to repair time. Hence, to transfer fuse data, the number of cycles required is at least equal to the number of e-memories on a die (e.g. nine cycles in Fig. 1). As a result, the larger the number of e-memories is on a die, the longer the repair time is. Therefore, for advanced SoC products with a large number of e-memories on a die, fuse schemes that are both area efficient and time efficient are essential.

3. Proposed Fuse Scheme

3.1 Description of Proposed Fuse Scheme

We propose a new fuse scheme that enables decreasing the fuse area and repair time of advanced SoC products.

Figure 2 shows a simplified block diagram of our proposed fuse scheme. As shown in Fig. 2, nine different types of e-memories with varying sizes are scattered on a die with two virtual e-memory groups comprising several e-memories. The virtual e-memory group *i* consists of six e-memories and each e-memory has a spare column in an IO unit. The virtual e-memory group *ii* consists of three e-memories and each e-memory has a spare IO. The virtual

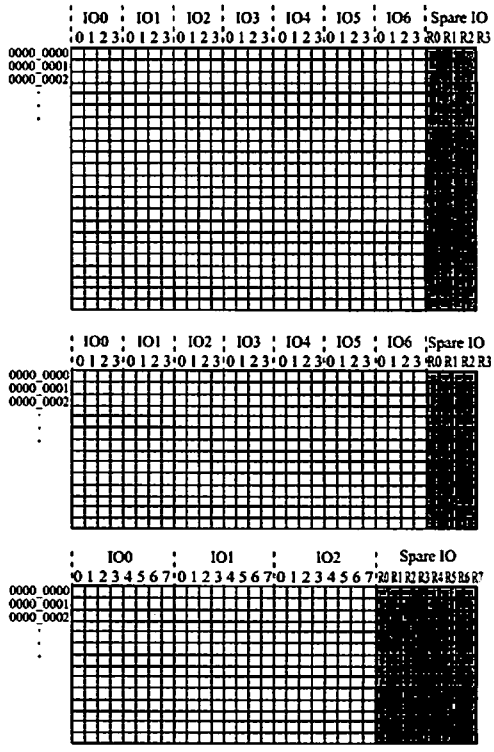


Fig. 3 Structure of e-memories in the virtual e-memory group *ii*.

e-memory groups *i* and *ii* are connected to different fuses, and e-memories belonging to the same virtual e-memory group are connected in parallel to the same fuse. Figure 3 shows the structure of e-memories in the virtual e-memory group *ii*. As shown in Fig. 3, each row, column, and IO has an address. When a failure bit is detected in an IO of an e-memory, both the IO address of the e-memory and the corresponding IO addresses of the other e-memories are si-

Table 1 Comparisons of fuse schemes.

Fuse schemes	Fuse architecture	Total fuse area	Repair time	Repair rate
Conventional [12]–[14]	Dedicated	Large	Short	Best
Conventional [10]	Compressed	Small	Long	Best
Conventional [11]	Shared	Small	Long	Best
Proposed	Partly shared	Small	Short	Almost Best

multaneously switched to spare addresses. For instance, if a *failure* occurs in IO5 in the upper e-memory in Fig. 3, the both IO5s in the upper and middle e-memories are simultaneously switched to spare IOs. On the other hand, switching does not occur in the lower e-memory because IO5 is absent in the lower e-memory. The virtual e-memory groups *i* and *ii* can be repaired independently and in parallel. As a result, in our proposed fuse scheme, fuse data transfer is achieved in merely one cycle.

Table 1 compares the features of conventional fuse schemes and our proposed fuse scheme. In the fuse schemes of [12]–[14], the number of fuses embedded in a die is the same as e-memories, whereas in our proposed fuse scheme and the fuse schemes of [10], [11], only several fuses (less than the number of e-memories) are embedded on a die. Therefore, compared with the fuse schemes of [12]–[14], our proposed fuse scheme and the fuse schemes of [10], [11] are effective in reducing the total fuse area on a die. Moreover, compared with the fuse schemes of [10], [11], our proposed fuse scheme is effective in reducing repair time because it requires less number of cycles for fuse data transfer. In particular, our proposed fuse scheme requires only one cycle to transfer fuse data, whereas the number of cycles required for the fuse schemes of [10], [11] is at the very least equal to the number of e-memories on a chip. However, the repair rate of our proposed fuse scheme is generally lower than that of the fuse schemes of [10], [11] because a fuse is sheared within a virtual e-memory group in the proposed fuse scheme. For example, when more than two failures occur in a virtual e-memory group, the chip cannot be repaired even if the number of virtual e-memory groups (i.e. the number of fuses) in the chip is more than the total number of failures. On the other hand, the chip can be repaired in the case that the number of fuses in a chip is more than the total number of failures in the conventional fuse schemes of [10], [11]. Moreover, in our proposed fuse scheme, repair rate is affected by the manner in which virtual e-memory groups are constructed. In the next section, we describe our strategy of grouping e-memories.

3.2 Optimizing Virtual e-Memory Groups

In our proposed fuse scheme, the number of virtual e-memory groups is optimized based on a conventional statistical yield model.

According to the conventional statistical yield model [15]–[19], the probability $P_{IO}(x)$ of x IOs failures occurring is given by

$$P_{IO}(x) = \binom{l+s}{x}(1-Y_{IO})^x \times Y_{IO}{}^{(l+s-x)}, \quad (1)$$

where l is the number of IOs in the main memory portion, s is the number of spare IOs in the redundant portion, and Y_{IO} is the yield of one IO, approximated by

$$Y_{IO} = \exp(-D \times A_{IO}). \quad (2)$$

In the above equation, D is the defect density and A_{IO} is the area of one IO. According to the statistical yield model, A_{IO} and D affect the probability of failures $P_{IO}(x)$. Equations 1 and 2 suggest that the total area of e-memories in a virtual e-memory group affects the probability of failures of the e-memories in the virtual e-memory group because the defect density D of the e-memories is generally almost constant during manufacturing.

In our proposed fuse scheme, e-memories in a virtual e-memory group share a fuse, which is used to switch a single address to a spare address. Therefore, it is impossible to repair more than two failures in the virtual e-memory group. When the total area of e-memories in a virtual e-memory group is large (i.e. the number of virtual e-memory groups is small), the probability of more than two failures occurring in the virtual e-memory group is high. Thus, the yield is low. On the other hand, when the number of virtual e-memory groups is large, the total fuse area becomes large because each virtual e-memory group requires its own fuse. Furthermore, the areas of the spare bit, the BIST and BISR circuits of an e-memory change according to the spare type (e.g. row, column, or IO). Therefore, to maximize the number of good chips on a wafer, the number of virtual e-memory groups must be optimized by considering a trade-off between reduction in the total die area and repair rate.

An objective function must be defined for optimizing the number of virtual e-memory groups. We adopt the following objective function to evaluate the trade-off between reduction in the total die area and repair rate.

$$X(n) = \frac{Y_{memGr}/Y_{mem_w/o\text{-}rep}}{A_{memGr}/A_{mem_w/o\text{-}rep}}, \quad (3)$$

where Y_{memGr} is the yield of product that has been repaired with its e-memories grouped into n virtual e-memory groups, $Y_{mem_w/o\text{-}rep}$ is the yield of a product without repairs, A_{memGr} is the area of the product with grouped e-memories, and $A_{mem_w/o\text{-}rep}$ is the area of the product without repairs. Y_{memGr} and A_{memGr} are approximated by Eqs. (4) and (6).

$$Y_{memGr} = \prod_{\alpha=1}^{n} Y_{memGr}(\alpha) \times Y_{logic}, \quad (4)$$

where

$$Y_{memGr}(\alpha) = \prod_{k=1}^{m} P_{bit(k)}(0)$$

$$+ Y_{\text{fuse}} \sum_{j=1}^{m} P_{\text{sparebit}(j)}(0) \times \prod_{k=1}^{m} P_{\text{bit}(k)}(\delta_{j,k}) \quad (5)$$

and

$$A_{\text{memGr}} = \sum_{\alpha=1}^{n} \sum_{\beta=1}^{m} (A_{\text{mem}}(\beta, \alpha) + A_{\text{sparebit}}(\beta, \alpha)) + \sum_{\alpha=1}^{n} A_{\text{fuse}}(\alpha) + A_{\text{logic}}. \quad (6)$$

In Eq. (4), Y_{logic} represents the yield of a logic circuit that includes BIST and BISR circuits and $Y_{\text{memGr}}(\alpha)$ is the yield of the virtual e-memory group α. In Eq. (5), $P_{\text{bit}(k)}(0)$ represents the probability that any bits in a row, column, or IO unit of an e-memory k of the virtual e-memory group α would not fail, Y_{fuse} is the yield of the fuse of the virtual e-memory group α, $P_{\text{sparebit}(j)}(0)$ is the probability that the spare bit of the e-memory j that is part of the virtual e-memory group α would not fail, and $P_{\text{bit}(k)}(\delta_{j,k})$ represents the probability that an arbitrary bit of an e-memory k in the virtual e-memory group α would fail or not, where $\delta_{j,k}$ is the Kronecker delta ($\delta_{j,k} = 0$ or 1). In Eq. (6), A_{logic} is the area of the logic circuit including the BIST and BISR circuits, $A_{\text{mem}}(\beta, \alpha)$ is the area of the e-memory β that is part of the virtual e-memory group α, $A_{\text{sparebit}}(\beta, \alpha)$ is the area of the spare bits of the e-memory β of the virtual e-memory group α, and $A_{\text{fuse}}(\alpha)$ is the area of the fuse of the virtual e-memory group α.

$Y_{\text{mem_w/o-rep}}$ and $A_{\text{mem_w/o-rep}}$ can be approximated as follows:

$$Y_{\text{mem_w/o-rep}} = \exp(-D \times A_{\text{mem_w/o-rep}}) \quad (7)$$

and

$$A_{\text{mem_w/o-rep}} = A_{\text{mem}} + A_{\text{logic_w/o-rep}}, \quad (8)$$

where A_{mem} represents the total area of the memory of the product without the spare bits and fuses, and $A_{\text{logic_w/o-rep}}$ is the area of the logic circuit without the BIST and BISR circuits.

Figure 4 illustrates the process used to optimize the number of virtual e-memory groups. First, the number of virtual e-memory groups n is set to 1. Second, one ($n = 1$) virtual e-memory group is created. Next, the objective function $X(n)$ (= $X(1)$) is computed. Because $n = 1$, after n is incremented by 1 ($n = 2$), the procedure returns to the second step in the flowchart. Hence, two ($n = 2$) virtual e-memory groups are created while maintaining approximately the same area for e-memories followed by the computation of $X(n)$ (= $X(2)$). The process is repeated until $X(n-1)$ becomes larger than $X(n)$, $n - 1$ is then chosen as the optimal number of virtual e-memory groups because the objective function $X(n)$ has the characteristic of monotonically increasing or decreasing depending on whether n is smaller or larger than the optimal number of virtual e-memory groups, respectively.

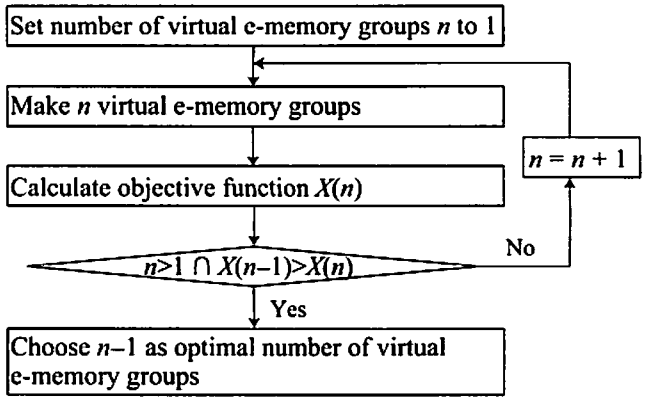


Fig. 4 Optimization of the number of virtual e-memory groups.

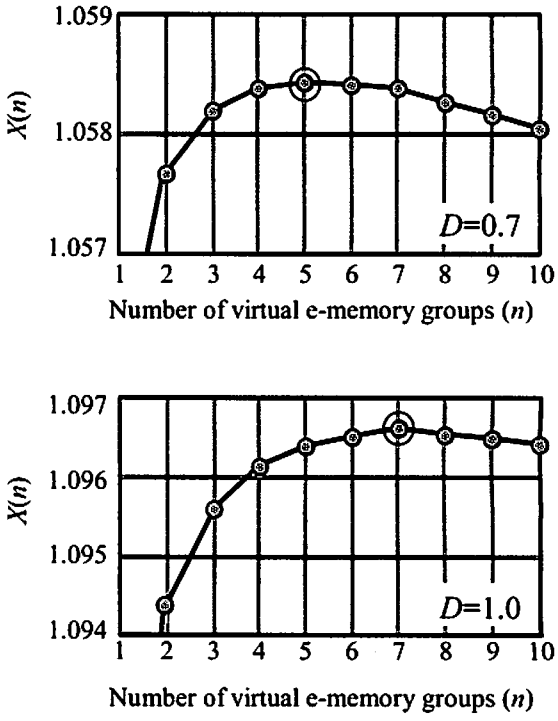


Fig. 5 Distribution of $X(n)$ (for 980 e-memories) at $D = 0.7$ (upper plot) and $D = 1.0$ (lower plot).

We present below an example of the process used to optimize the number of virtual e-memory groups. In our example, a die contains 980 e-memories of the same size. Each e-memory consists of 256 rows and 32 IOs, and each IO consists of 4 columns. Therefore, the total capacity of the e-memories is over 32 Mb. Moreover, each e-memory has a spare IO. Figure 5 shows the value of the objective function as n increases from 1 to 10. The upper and lower graphs plot the value of $X(n)$ when the defect density D is 0.7 and 1.0, respectively. As shown in Fig. 5, the objective functions in the upper and lower graphs are maximum at $n = 5$ and $n = 7$, respectively. These results indicate that the number of good

Table 2 Yield of each virtual e-memory group n for 980 e-memories.

n	1	2	3	4	5	6	7	8	9	10
D = 0.7	96.18%	96.36%	96.42%	96.45%	96.47%	96.48%	96.49%	96.50%	96.51%	96.51%
		(0.18%)	(0.06%)	(0.03%)	(0.02%)	(0.01%)	(0.01%)	(0.00%)	(0.01%)	(0.00%)
D = 1.0	94.46%	94.80%	94.92%	94.98%	95.02%	95.04%	95.06%	95.07%	95.08%	95.09%
		(0.34%)	(0.12%)	(0.06%)	(0.04%)	(0.02%)	(0.02%)	(0.01%)	(0.01%)	(0.01%)

dies per wafer is maximized when the number of virtual e-memory groups is five and seven. In both cases, the objective functions monotonically increase or decrease depending on whether n is smaller or larger than the optimal number of the virtual e-memory groups, respectively. Table 2 shows the yield (Y_{memGr}) after repair as the number of e-memory groups n increases from 1 to 10. The values in parentheses are the differences between Y_{memGr} with $n+1$ and n virtual e-memory groups. These differences become small as n becomes large. The Y_{memGr} values of the proposed fuse scheme with an optimal number of virtual e-memory groups and the conventional schemes in [10], [11] are 96.47% and 96.55% (D = 0.7), and 95.06% and 95.16% (D = 1.0), respectively. In other words, the yield of the proposed fuse scheme is only 0.08% (D = 0.7) and 0.10% (D = 1.0) lower than that of the conventional fuse schemes.

In the proposed fuse scheme, the number of cycles required to transfer fuse data is 1, whereas the conventional fuse schemes of [10], [11] require at least 980 cycles. The reduction in the number of cycles means a reduced fuse data transfer time which is a part of repair time. That is, the proposed fuse scheme can dramatically reduce fuse data transfer time. In this case, the proposed fuse scheme achieves 99.9% reduction in the fuse data transfer time compared with the conventional fuse schemes. On the other hand, almost the same area of fuses are necessary in the proposed fuse scheme and the conventional fuse schemes of [10], [11] because the number of fuses required on a die is almost the same in these schemes.

In summary, our proposed fuse scheme reduces both fuse area and repair time without any serious yield loss and therefore is effective especially for advanced SoC products that have a large number of small e-memories on a die.

4. Verification of Efficiency of Proposed Fuse Scheme

To verify the effectiveness of our proposed fuse scheme, we applied it to a real 65 nm SoC product in which each e-memory has a spare IO. As discussed in the previous section, the number of good dies on a wafer can be maximized by optimizing the number of virtual e-memory groups. However, the variation in defect density D throughout the lifetime of a product should be considered when computing $X(n)$. As the manufacturing process of SoC products matures, D decreases [20]–[22]. Therefore, the optimal number of virtual e-memory groups can be obtained if the number of good dies is maintained at maximum throughout the lifetime of the SoC product. Hence, the number of the virtual e-memory groups is optimized by maximizing $X(n)$ using time-varying D.

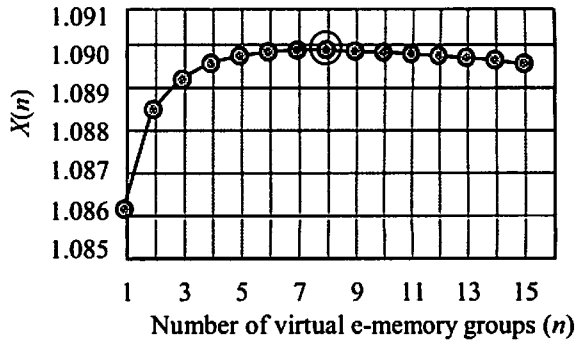


Fig. 6 Distribution of $X(n)$ for the 65 nm SoC product.

Figure 6 shows the calculation result of $X(n)$ for time-varying D. In this calculation, to account for the time-varying nature of D, we use the average value of D throughout the lifetime of the product. This average value is computed as a weighted mean of the ratio of the monthly average of D and the monthly production volume. Here, the monthly average of D is determined on the basis of our manufacturing experience of other products and the monthly production volume is determined on the basis of the production plan of the 65 nm SoC product. The maximum value of $X(n)$ is obtained when n = 8. The average of the yield throughout the lifetime of the 65 nm SoC product with eight virtual e-memory groups is only 0.06% lower than that of the conventional fuse schemes in [10], [11]. This difference is statistically insignificant because yield variation with lots or wafers is much larger. The fuse area of the proposed fuse scheme is almost the same as the conventional fuse schemes of [10], [11]. Therefore, the number of good chips on a wafer of the proposed fuse scheme is almost the same as the conventional fuse schemes of [10], [11]. On the other hand, the reduction in the fuse data transfer time is over 99.0% compared with the conventional fuse schemes because several hundreds of e-memories are embedded on the products. These results demonstrate that the proposed fuse scheme is effective in improving the number of good chips on a wafer with a short repair time.

5. Conclusions

In advanced SoC products with thousands of small e-memories on a die, fuse area reduction is an effective technique to maximize the number of good chips per wafer. Although there are several approaches available for fuse area reduction, they result in an increased repair time because they require more cycles to transfer fuse data. In this paper, we proposed a new fuse scheme that e-memories are orga-

nized into virtual e-memory groups. We showed that our proposed scheme reduces both fuse area and repair time by optimizing the number of virtual e-memory groups. Next, we verified our proposed fuse scheme by applying it to a 65 nm SoC product. Although our proposed scheme causes a 0.06% reduction in the yield, it achieves over 99.0% reduction in the fuse data transfer time compared with the conventional fuse schemes of [10], [11]. This result proves that our proposed fuse scheme can effectively increase the number of good dies in advanced SoC products while achieving a short repair time.

In this paper, we presented applications in which each e-memory had a spare IO. However, in our proposed fuse scheme, each e-memory can have a spare row or column. Our future studies will focus on the proposal of grouping and optimizing methods for various spare types of e-memories (e.g. row, column, and/or IO).

Acknowledgments

We would like to thank Mr. O. Tada, Mr. W. Uchida, Mr. H. Maeno, and Mr. K. Wakabayashi of Renesas Electronics Corporation, for their invaluable discussions. We would also like to thank Mr. M. Sawa, Dr. K. Funaki, Dr. T. Miwa, and Mr. J. Nakazato of Hitachi's Yokohama Research Laboratory (YRL) for their comments and encouragement.

References

[1] Y. Zorian, "Embedded-memory test & repair: Infrastructure IP for SOC yield," IEEE Trans. Design & Test of Computers, vol.20, no.3, pp.58–66, May 2003.

[2] X. Wang, M. Ottavi, F.J. Meyer, and F. Lombardi, "Estimating the manufacturing yield of compiler-based embedded SRAMs," IEEE Trans. Semicond. Manuf., vol.18, no.3, pp.412–421, Aug. 2005.

[3] H. Melzner and A. Olbrich, "Maximization of good chips per wafer by optimization of memory redundancy," IEEE Trans. Semicond. Manuf., vol.20, no.2, pp.68–76, May 2007.

[4] R.F. Huang, J.F. Li, J.C. Yeh, and C.W. Wu, "Raisin: Redundancy analysis algorithm simulation," IEEE Design & Test of Computers, vol.24, no.4, pp.386–396, July-Aug. 2007.

[5] B. Wang and Q. Xu, "Test/Repair area overhead reduction for small embedded SRAMs," Proc. 15th Asian Test Symposium, pp.37–44, Nov. 2006.

[6] V. Schober, S. Paul, and O. Picot, "Memory built-in self-repair using redundant words," Proc. International Test Conference, pp.995–1001, Oct. 2001.

[7] Y. Zorian, "Embedded memory test & repair: Infrastructure IP for SOC yield," Proc. International Test Conference, pp.340–349, Oct. 2002.

[8] R.C. Aitken, "A modular wrapper enabling high speed BIST and repair for small wide memories," Proc. International Test Conference, pp.997–1005, Oct. 2004.

[9] T.W. Tseng and J.F. Li, "A shared parallel built-in self-repair scheme for random access memories in SOCs" Proc. International Test Conference, pp.1–9, Oct. 2008.

[10] M.R. Ouellette, D.L. Anand, and P. Jakobsen, "Shared fuse macro for multiple embedded memory devices with redundancy," Proc. Custom Integrated Circuits, pp.191–194, May 2001.

[11] A. Garg and P. Dubey, "Fuse area reduction based on quantitative yield analysis and effective chip cost" Proc. International SOC Conference, pp.24–27, Sept. 2006.

[12] S. Kikuda, H. Miyamoto, S. Mori, M. Niiro, and M. Yamada, "Optimized redundancy selection based on failure-related yield model for 64-Mb DRAM and beyond," IEEE J. Solid-State Circuits, vol.26, no.11, pp.1550–1555, Nov. 1991.

[13] T. Kirihata, Y. Watanabe, H. Wong, J. DeBrosse, M. Yoshida, D. Katoh, S. Fujii, M. Wordeman, P. Poechmueller, S. Parke, and Y. Asao, "Fault-tolerant designs for 256 Mb DRAM," IEEE Proc. VLSI Circuits, pp.107–108, June 1995.

[14] T. Yamagata, H. Sato, K. Fujita, Y. Nishimura, and K. Anami, "Distributed globally replaceable redundancy scheme for sub-half-micron ULSI memories and beyond," IEEE J. Solid-State Circuits, vol.31, no.2, pp.195–201, Feb. 1996.

[15] C.H. Stapper, A.N. McLaren, and M. Dreckmann, "Yield model for productivity optimization of VLSI memory chips with redundancy and partially good product," IBM J. Research Development, vol.24, no.3, pp.398–409, 1980.

[16] J. Khare, D.B.I. Feltham, and W. Maly, "Accurate estimation of defect-related yield loss in reconfigurable VLSI circuits," IEEE J. Solid-State Circuits, vol.28, no.2, pp.146–156, Feb. 1993.

[17] J.D. Segal, S. Bakarian, J.E. Colburn, M. Kumar, C. Hong, and A. Shubat, "Determining redundancy requirements for memory arrays with critical area analysis," Proc. IEEE International Workshop on Memory Technology, Design and Testing, pp.48–53, Aug. 1999.

[18] D.J. Ciplickas, X. Li, and A.J. Strojwas, "Predictive yield modeling of VLSIC's," Proc. Fifth International Workshop on Statistical Meteorology, pp.28–37, June 2000.

[19] Y. Hamamura, K. Nemoto, T. Kumazawa, H. Iwata, K. Okuyama, S. Kamohara, and A. Sugimoto, "Repair yield simulation with iterative critical area analysis for different types of failure," Proc. 17th IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems, pp.305–313, Nov. 2002.

[20] W. Maly, "Computer-aided design for VLSI circuit manufacturability," Proc. IEEE, vol.78, no.2, pp.356–392, Feb. 1990.

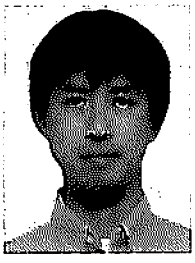
[21] T. Chen and M. Wang, "A fuzzy set approach for yield learning modeling in wafer manufacturing," IEEE Trans. Semicond. Manuf., vol.12, no.2, pp.252–258, May 1999.

[22] S. Simmons, "Modeling yield throughout the DRAM product life cycle," Proc. IEEE International Symposium on Semiconductor Manufacturing Conference, pp.123–126, Oct. 1999.

Chizu Matsumoto received B.S. and M.S. degrees in marine bio-resource and environment sensing from Hokkaido University, Hokkaido, Japan, in 2002 and 2004, respectively. She joined the Production Engineering center of Yokohama Research Laboratory, Hitachi, Ltd. Yokohama, Japan, in 2004. She is currently working for the Manufacturing 1st Department laboratory. She has been engaged in defect based yield management methodologies and systems. Her research interests include yield management and enhancement and digital system testing.

Yuichi Hamamura (M'02) received B.S. and M.S. degrees in mechanical engineering from Tokyo Institute of Technology, Japan, in 1989 and in 1991, respectively, and then a Ph.D. degree in information science from Hokkaido University, Japan, in 2008. He joined the Production Engineering center of Yokohama Research Laboratory, Hitachi, Ltd., Yokohama, Japan, in 1991. He was engaged in the development of focused ion beam systems and yield management system. He spent a year at Georgia Institute of Technology, Atlanta, for cooperative research. He is now a general manager of Yokohama Research Laboratory. He is a member of the IEEE.

Michinobu Nakao received the B.S. and M.S. degrees in mathematics from Kyushu University, Japan, in 1989 and 1991, respectively, and then a Ph.D. degree in electrical engineering from Tokyo Metropolitan University, Japan, in 2003. He joined Hitachi, Ltd in 1993. He is now a senior professional of Renesas Electronics Corporation. His research interests include design for test and fault diagnosis.



Kaname Yamasaki received the B.E. degree in electrical and electronic engineering and the M.E. degree in physical electronics from Tokyo Institute of Technology, Tokyo, Japan, in 1991 and 1993, respectively. He joined the Central Research Laboratory, Hitachi Ltd., Tokyo, Japan in 1993. He has been engaged in the research and development of high-speed memories for 8 years. He has been concerned with test methodology for memory in SoC and SiP since 2001. He is currently a senior professional in the design-for-test technology development department of Renesas Electronics Corp., Tokyo, Japan.

Yoshikazu Saito received the B.S. degree from Nara College of Technology, Nara, Japan, in 1982. He joined the Semiconductor & Integrated Circuits, Hitachi, Ltd., Tokyo, Japan, in 1982. Since 2003, he has been a member of Renesas Electronics Corp., Tokyo, Japan. He has been engaged in the embedded memory design.

Shun'ichi Kaneko received the B.S. degree in precision engineering and the M.S. degree in information engineering from Hokkaido University, Japan, in 1978 and 1980, respectively, and then the Ph.D. degree in systems engineering from the University of Tokyo, Japan, in 1990. He had been a research assistant of the Department of Computer Science since 1980 to 1991, an associate professor of the Department of Electronic Engineering since 1991 to 1995, and an associate professor of the Department of Bio-application and Systems Engineering since 1995 to 1996, in Tokyo University of Agriculture and Technology, Japan. He is a professor of the Laboratory of System Sensing and Control in the Graduate School of Information Science and Technology in Hokkaido University. He received the Best Paper Award in 1990, the Society Award in 1998, respectively, from Japan Society of Precision Engineering. His research interest includes machine vision, image sensing and understanding, robust image registration. He is a member of JSPE, IEEJ, SICE and the IEEE computer society.