Efficient and Low-Cost Metal Revision Techniques for Post Silicon Repair

Sungchul Lee and Hyunchul Shin

Abstract—**New effective techniques to repair "small" design errors in integrated circuits are presented. As semiconductor chip complexity increases and the design period becomes tight, errors frequently remain in a fabricated chip making revisions required. Full mask revision significantly increases the cost and time-to-market. However, since many "small" errors can be repaired by modifying several connections among the circuit blocks and spare cells, errors can frequently be repaired by revising metal layers. Metal only revision takes significantly less time and involves less cost when compared to full mask revision, since mask revision costs multi-million dollars while metal revision costs tens of thousand dollars. In our research, new techniques are developed to further reduce the number of metal layers to be revised. Specifically, we partition the circuit blocks with higher error probabilities and extend the terminals of the signals crossing the partition boundaries to the preselected metal repair layers. Our partitioning and pin extension to repair layers can significantly improve the repairability by revising only the metal repair layers. Since pin extension may increase delay slightly, this method can be used for non-timing-critical parts of circuits. Experimental results by using academia and industrial circuits show that the revision of the two metal layers can repair many "small" errors at low-cost and with short revision time. On the average, when 11.64% of the spare cell area and 24.72% of the extended pins are added to the original circuits, 83.74% of the single errors (and 72.22% of the double errors) can be corrected by using two metal revision. We also suggest methods to use our repair techniques with normal commercial vender tools.**

Index Terms—**Repair, post-silicon, pin extension, partition, revision**

I. INTRODUCTION

In modern design, pre-silicon validation becomes quite difficult because of the increased functions and more complex process development. If errors are discovered after chip fabrication, revision is necessary. First silicon debugging requires a labor-intensive engineering effort of many months and it has become the most time-consuming part (35% on average) of the development cycle of a new chip [1]. The first silicon IC failure rate was reported to be higher than 70% [2]. Full revision including re-design and re-fabrication involves a large cost and causes substantial delay in product development. For example, full mask cost of 65nm technology costs close to 3 million dollars and 32nm technology costs in the range of 6 million dollars [3] while metal revision costs tens of thousand dollars only. Therefore, effective techniques are essential to fix errors found post-silicon. For partial revision, spare cells are fabricated and used to replace the erroneous parts during post-silicon error repair.

The spare cells can be fixed cells like standard cells or flexible reconfigurable cells. The repair method with standard cells has the advantage when the error can be repaired by replacing a small number of standard cells. In

Manuscript received Dec. 22, 2013; accepted Apr. 24, 2014
Department of Electronical Communication Engineering, Hanyang University, Ansan, Korea
E-mail : shin@hanyang.ac.kr

cases of several errors or a “huge” error, repair by using standard spare cells can be worse than full revision in terms of wirelength and delay. Repair methods with reconfigurable cells can implement several different functions by configuring cells with overhead in area and delay.

For interconnection revision, re-routing is performed either by a programmable interconnect or by metal layer revision. A standard metal method revises only one via layer to re-route for repair [4]. As shown in Fig. 1, ViaPath programmed vias are used to modify interconnects by using the vias between metal 3 and 4 layers. Spare cell placement methods are proposed in [5].

In the Fogclear [6] method, functional errors and electrical errors are repaired by using a focused Ion Beam (when small changes can fix the errors) or by using full revision. [7] showed techniques to re-synthesize logic cones within a large design impacted by multiple RTL changes in order to accommodate a late functional engineering change order (ECO). Cone re-synthesis flow [7] has 2 parts: (a) a front end subsystem that generates the logic level netlist that is formally equivalent to the ECO’ed RTL and (b) a back end subsystem that places the newly added cells, and incrementally routes the corresponding nets with a minimum disturbance to the majority of the nets in the design. However, post silicon repair is very limited and full mask revision involves a large cost, usually multi-million dollars, for modern technology [8].

In this paper, we describe new fast and low-cost techniques to repair design errors which can be repaired by changing several connections among the circuit blocks and spare cells. New silicon chips are now designed by integrating a number of circuit blocks. Most of these circuit blocks are reused from other previous designs, and some of them are new blocks. Frequently, design errors are introduced in the new blocks since the reused blocks have usually been verified. When designers can identify more error-prone circuit blocks, we may allocate more repair resources around the blocks so that the errors in the blocks, if any, can be repaired efficiently. Full revision may take several million dollars and a couple of months. Metal only revision, in which all metal layers are redesigned to change the interconnection, takes only a fraction of the cost and revision time. If one can repair by modifying two metal layers, the revision cost and the revision time can be significantly reduced even further. As far as authors know, this is the first report on design repair by using partial metal layer revision.

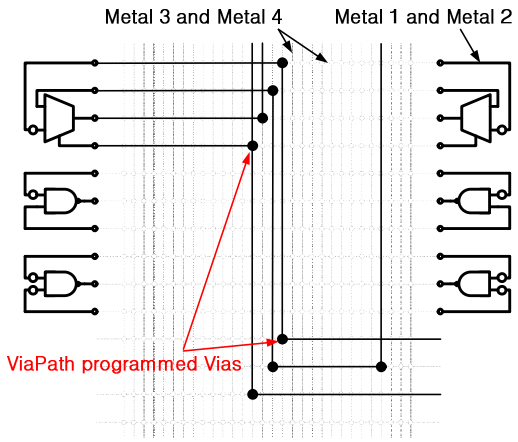


Fig. 1. Interconnection fabric.

In Section II, the new fast and low cost repair techniques are described. Currently we revise two metal layers to repair the errors if possible. In Section III, a partitioning technique is described to minimize the revision area. In Section IV, new pseudo-cells are proposed to use our repair techniques with normal commercial placement and routing tools. In Section V, the experimental results are explained. Finally, the conclusions are summarized in Section VI.

II. Fast and Low-Cost Post-Silicon Repair

When an error is found after fabrication, an efficient repair method can be chosen to correct the error. Repair by using programmable vias on a single via layer as in [4] is a simple way. In this standard metal method, crossbar type switches are formed using a single layer of vias. However, the cross-bar style interconnecting structure requires a large area and delay overhead. Programmable spare cells can reduce mesh crossbar type metal wires. However, programmable cells are large and need control signals. Furthermore, a programmable cell usually has a single output and it is costly to build a multiple output function. To allow for greater flexible routing capability, we adopt a two metal layer revision method so that one horizontal layer and one vertical routing layer are available for repair. New techniques are

developed to improve the repairability. Fully verified circuit blocks are usually free from errors, while newly designed or modified circuit blocks are rather susceptible to errors. When the signals from more "error prone" circuit blocks are located through the repair layers, they can be re-routed by revising only the repair layers during the post-silicon repair.

When an error cannot be repaired by revising the repair layers, full metal layer revision or even full mask revision may be required. When an error is found, the correct circuit block should be synthesized by using spare cells. Then the erroneous block is isolated and a new synthesized block is connected by rerouting through the repair metal layers. Fig. 2 shows an example.

We assume circuit block B is in error and block C is the spare block which can implement the correct function. Metal layers 3 and 4 are the repair layers. The erroneous circuit in Fig. 2(a) can be repaired by re-routing the repair layers to disconnect the connections to block B and to make new connections to block C, as shown in Fig. 2(b).

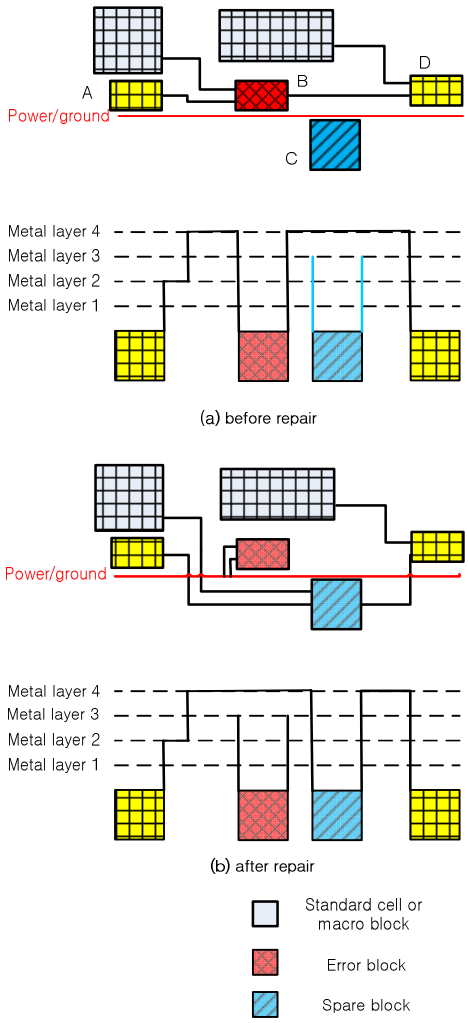


Fig. 2. Repair by two metal layer revision.

To make the repair possible by revising only the repair layers, all the signals of the erroneous blocks have to go through the repair layers so that they can be disconnected and re-routed. Furthermore, the repair block should be able to be constructed by using the spare cells, and their external signals have to be available to the repair layers. Therefore we extend the terminals of the error-prone blocks and the spare cells to the repair layers. If the number of terminals extended to the repair layers increases, the repairability increases and the overhead in the resource also increases. But this technique is focused on design revision for error-prone new blocks. Most of circuit blocks are verified and reused from other previous designs and new block size is small, so overhead from pre-establishment will not be large.

Repair, if necessary, can be performed as circuit blocks or clusters. "Good" partitioning can minimize the repair cost by reducing the repair block size.

III. Partitioning for Repairability

When the size of the error block to be replaced is small and the size of the new synthesized block to replace the error block is small, the repairability improves. Fig. 3 shows an example. In Fig. 3(a), an error block S is found and the whole block S is replaced by the repair block RL for repair as shown in Fig. 3(b). The repair cost (resource) can be reduced, however, when the circuit blocks are partitioned into sub-blocks. For example, when S is partitioned into S1, S2, S3, and S4 and the error can be localized in a sub-block for example S3, then only S3 can be replaced by a smaller repair block RS, as shown in Fig. 3(d)

In our current approach, the partitioning is performed based on the following information. First, we use the hierarchical information given by the designer. Second, we partition each block into a sequential (memory, register, flip-flop) part and a combinational part. Since the sequential components have timing issues like clock skew and are frequently larger in size than the combinational components (logic gates), it is advantageous to reuse the sequential parts during repair.

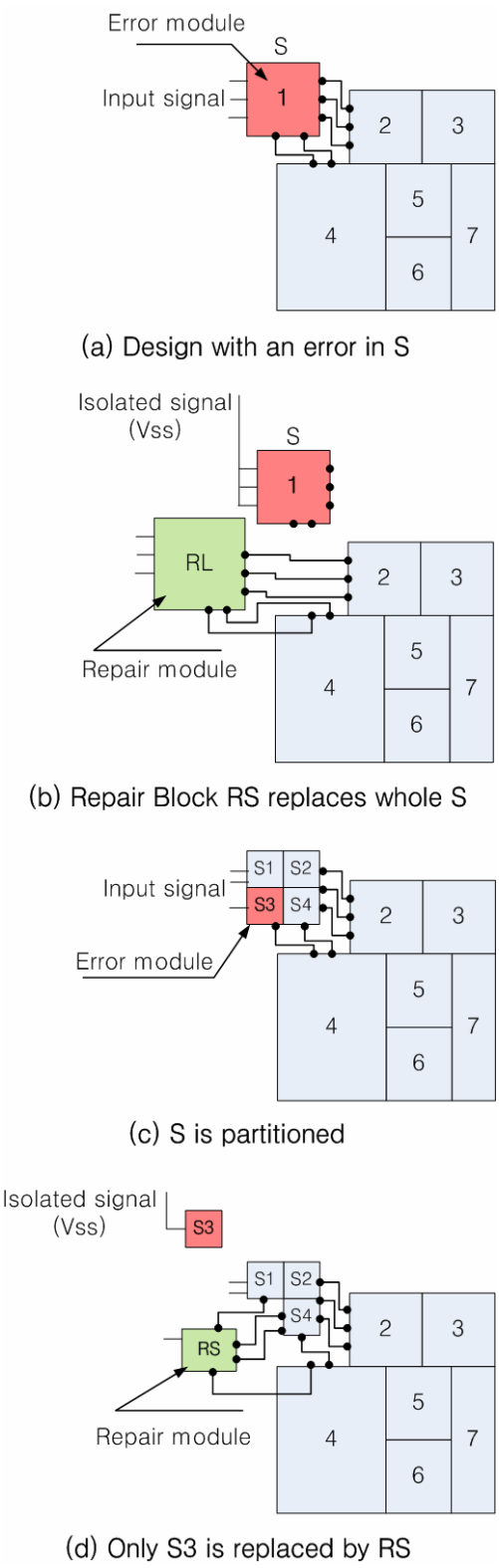


Fig. 3. Repair with spare cells.

Third, the combinational parts bigger than a threshold are partitioned into small sub-blocks. Fig. 4(b) shows the results of simple partitioning of the circuit from Fig. 4(a).

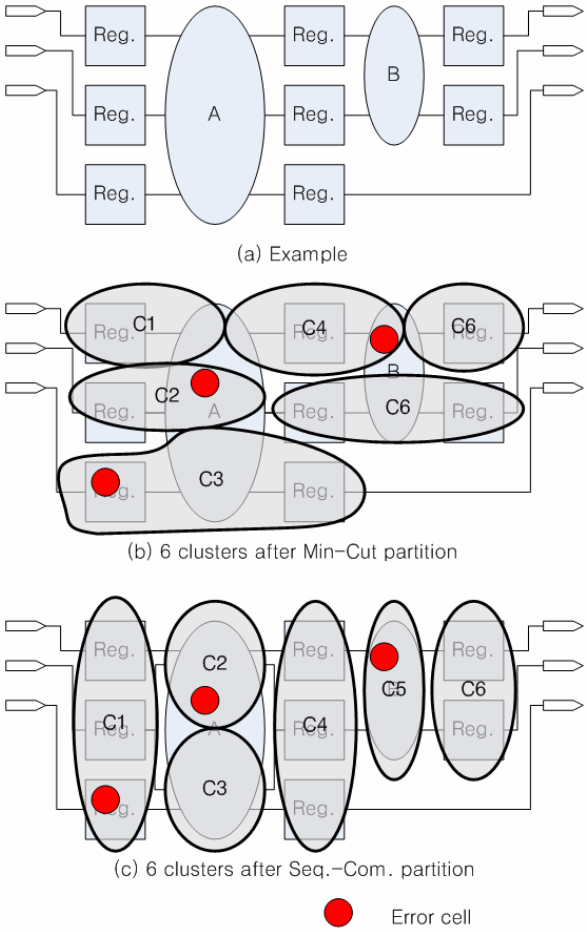
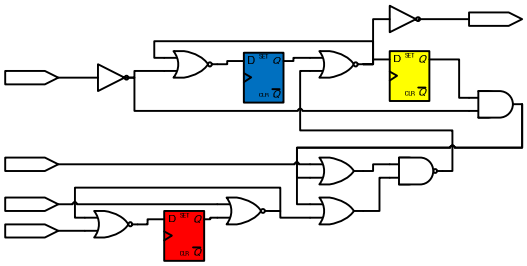


Fig. 4. Sequential-combinational partition (with 3 errors).

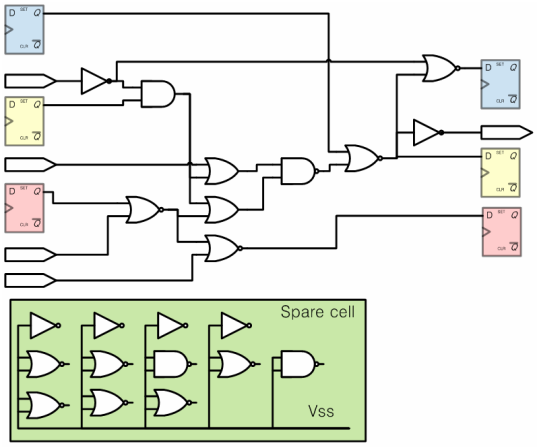
Fig. 4(c) shows our partitioning of the sequential and combinational parts. All the terminals crossing the partition boundaries are extended to the repair layers so that they can be re-routed during repair when needed.

Since sequential spare cells are separately prepared compared to combinational spare cells, the separation of the sequential parts and the combinational parts is very useful during the repairing process. Furthermore, when we extend the I/O signals of the sequential parts to the repair layers, the separation can significantly reduce the circuit block size of the blocks to be replaced.

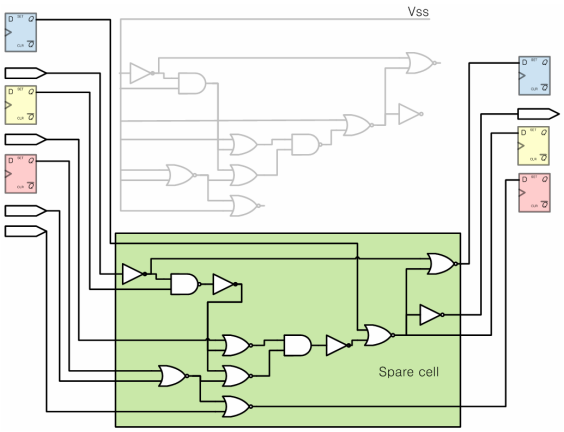
For example, Fig. 5 shows an example of repair for S27. Since Fig. 5(a) is a simple circuit, partition separates the combinational part and the three registers, as shown in Fig. 5(b). The I/O signals of the registers become extended pins that can be re-routed during repair. When there is a functional error in the combinational part, it can be replaced by spare cells forming a correct circuit, as shown in Fig. 5(c). The registers can be reused. When



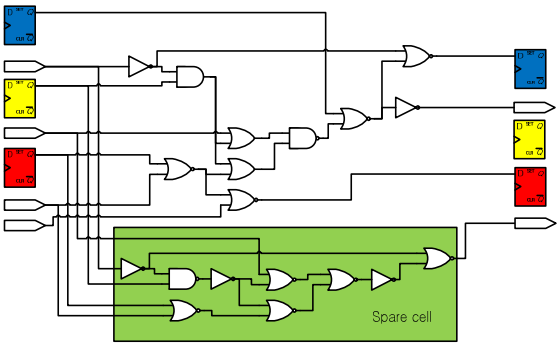
(a) Original logic



(b) Register pin extension to repair layers



(c) Repair of functional errors



(d) Repair of a missing function

Fig. 5. S27 example.

there is a missing function, it can also be implemented by spare cells forming a circuit to perform the missing function, as shown in Fig. 5(d). Sometimes, errors may not be repaired by using two metal layer revision. Then full metal revision or full mask revision should be used with high re-spin cost and period.

The granularity of the partitioning and the amount of spare cells are decided by the designer. Finding the optimal partitioning for good repairability under a given resource is a difficult problem since one cannot predict the errors. The designer may decide the partitioning and the amount of spare cells based on the overhead in area or delay. Since theoretical optimization is difficult, we have developed a new experiment-based optimization technique, for automatic design for repair.

First, we define the repair cost function to reduce the logic area and the routing resources as follows

$$\text{Repair\_cost}=(\text{total\_circuit\_area})\times(\text{total\_extended\_pins})$$

where the total_circuit_area is the sum of the original circuit area and spare circuit area, while the total_extended_pins is the total number of extended pins of the original circuit and the spare cells. This is to reduce the total circuit area and the routing resources. Now the partition granularity and the amount of the spare cells are determined by using the following exploring algorithm, as shown in Fig. 6.

For example, when Kp = 50, 50 different partitions are evaluated. If Nr = 3, we average the repair_cost of the three repair trials. If Ke=3, three random errors are introduced and repaired. [9], 8 error types are discussed, which are gate substitution, extra inverter, extra wire, extra gate, missing gate, missing inverter, multiple inverter, and extra wires. In our present implementation,

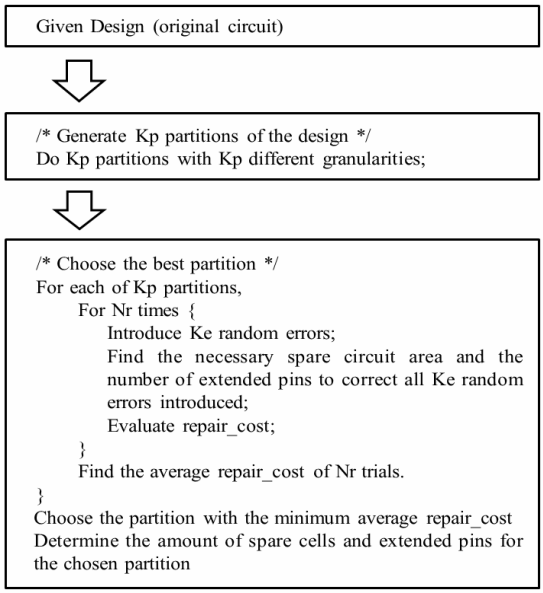


Fig. 6. Procedure for automatic design for repair.

Kp = 50 and 50 different partitions are evaluated. If Kp is large (fine granularity), the required spare cell area is usually small since a small partitioned part can be replaced. However, the number of extended pins will increase in general, since the fine granularity may cause a greater number of extended pins. Currently Nr = 3 is used to consider the variations depending on the errors. Ke can be varied. If Ke is large more errors can be fixed by using a larger repair cost. A designer can find an appropriate solution in the repairability and cost by adjusting the parameters, Kp, Nr, and Ke.

IV. Pseudo-Interconnection and Pseudo-Cell Techniques during Physical Design

In this section, spare cell placement techniques and terminal extension techniques are described. Spare cells can be uniformly preplaced or can be placed with other circuit blocks as in [5]. Since spare cells are not initially connected with other cells, we add the pseudo interconnections between the spare cells and the error-prone circuit blocks. Then a commercial placement tool can place spare cells near the error-prone blocks. The pseudo interconnections are removed after placement. We have devised our repair techniques to use popular commercial tools to avoid developing a new physical design tool as it may require a serious effort. After partitioning as described in the previous section, the

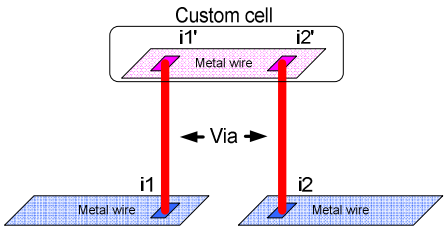


Fig. 7. Custom cell.

terminals crossing the partition boundaries need to be extended to the repair layers. In commercial placement tools, this extension function is not available at this time. Therefore, we developed a new pseudo custom cell approach. An example is shown in Fig. 7.

In the figure, i1 and i2 need to be connected through extended terminals on a repair layer so that they can be rerouted during repair if necessary. We add a pseudo custom cell on the repair layer. The pseudo-cell contains only a minimum sized wire segment with two terminals of net id i1, and i2. Then a placement tool will place the pseudo custom cells with other cells optimizing any given design objectives. The terminal i1 and i1' (i2 and i2') are to be connected during routing resulting in a complete connection of i1, i1', i2', i2. Design for repair can be applied only for a part of the circuit designed. When many IP blocks are reused and already verified, the spare cells and the extended pins can be added only for unproven risky parts.

V. Experimental Results

To repair design by metal layer revision, a faulty circuit is replaced by a correct circuit implemented by prefabricated spare cells on a chip. One may use either programmable cells or standard cells as spare cells. We compare the area and the number of vias to build a correct circuit (1) by using standard cells and (2) by using programmable cells based on the Via Patterned Gate Array (VPGA) [10]. Two metal layer revision is used for both cases.

Table 1 shows the repair experimental results for four benchmark circuits. Example S27 is from ISCAS89 and one of the two S27 circuits is replaced. In Bitsel, a bit selection circuit is added. In the Saturation circuit, the saturation function is added to the 4bit adder, In TM, a missing gate is added at the output stage. The cell area is

Table 1. Spare cell types : standard cells vs. VPGA programmable cells

Examples	Standard cell			VPGA programmable cells		
	Spare cell area (um2)	# of Vias		Spare cell area (um2)	# of Vias	
		In Logic	In Spare Cell		In Logic	In Spare Cell
S27	113.10	7	34	239.5(212%)	7	48
Bitsel	99.79	16	30	159.67(160%)	16	32
Saturation	66.53	4	24	718.50(480%)	4	64
TM	16.63	12	5	79.83(480%)	12	16
Sum	296.05	39	93	798.3(270%)	39	160

Table 2. Wire length results : Standard Metal vs. 2 Metal Revision

Examples				Standard cell & 2 Metal Revision			Structured ASIC & Standard Metal		
Name	Cells (area, um2)	Pins		Spare cells (area, um2)	Wirelength (um)		Spare cells (area, um2)	Wirelength (um)	
		Input	Output		Before repair	After repair		Before repair	After repair
S27	26 (545.53)	6	1	13 (113.10)	73.04	248.49	2 (419.13)	4048.88	4048.88
Bitsel	8 (186.28)	10	6	11 (99.79)	132.6	223	2 (419.13)	7307.5	7307.5
Saturation	4 (332.64)	8	4	8 (66.528)	9.2	105.7	1 (209.56)	818.4	818.4
TM	15 (781.70)	2	2	2 (16.63)	2.1	8.5	1 (209.56)	223.0	223.0
Sum	53 (1846.15)			34 (296.05)	216.94	585.69	6 (1257.38)	12397.78 (5714%)	12397.78 (2116%)

computed based on SMIC 0.18μm processing technology. A two level Mux tree is used for VPGA programmable cells. Table 1 shows that the standard cells use significantly less area and vias than those of the VPGA programmable cells. This is because programmable cells use a larger number of input ports.

Programmable cells [11] and standard metal [4] methods are used for fast chip fabrication. Even though this technique can also be used for design repair, it is not cost effective since crossbar switches by using the single via layer requires large area. For the same benchmark circuits used in Table 1 and for the same error corrections, two metal layer revision is significantly less costly than standard metal as shown in Table 2, both in the spare cell area and in the wirelength. Fig. 8 shows the layouts of the corrected S27 circuit using the same scale. In Fig. 8(b), the spare cells can be placed underneath the crossbar grids.

Since errors cannot be predicted, minimal amount of spare cells and extended pins to correct errors cannot be pre-computed. A procedure to decide the amount of spare cells and extended pins is shown in Fig. 6. When Kp = 50, Nr = 3, and Ke = 3, we obtained the results shown in Table 3. In the table, FSMA to FSMD are the industrial finite state machine circuits. AC97 controller (AC97),

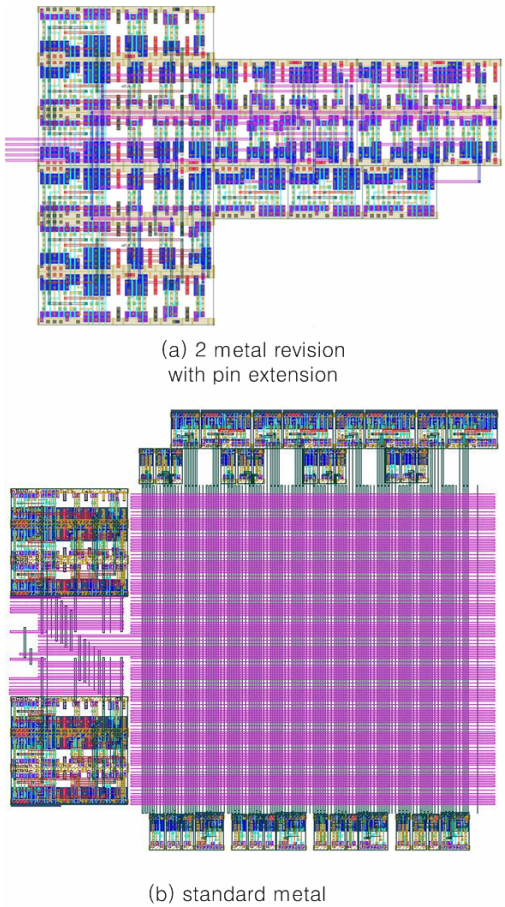


Fig. 8. Layouts: Standard metal vs. 2 metal revision.

Table 3. Repair rates

Examples	# of Cells	Cell area(um2)	Spare cell area(um2)	# of extended pins	Repair success rates (%)			
					1 Error	2 Errors	3 Errors	5 Errors
FSMA	37	1124.30	249.47 (22.19%)	79 (34.96%)	78.33	63.33	52.00	28.00
FSMB	135	4457.40	429.11 (9.63%)	87 (13.90%)	84.00	71.33	58.33	41.67
FSMC	118	3346.37	362.56 (10.83%)	96 (17.71%)	84.33	71.00	59.00	41.00
FSMD	173	4098.16	322.65 (7.87%)	92 (12.64%)	83.33	69.67	58.00	40.33
MD5	3934	193718.92	6502.99 (3.36%)	2928 (15.38%)	76.33	55.00	48.33	0.33
AC97	4879	233144.48	5874.20 (2.52%)	4288 (18.85%)	89.00	77.33	67.33	52.33
USB	5164	182808.03	6087.12 (3.33%)	6550 (30.52%)	80.00	63.33	51.00	26.00
Ethmac	26346	1041834.94	353326.66 (33.91%)	108677 (49.97%)	98.33	95.67	92.67	88.67
Reed_Solomon	35637	1209300.62	134502.92 (11.12%)	53209 (28.59%)	80.00	83.33	76.33	62.00
Average			11.64%	24.72%	83.74	72.22	62.55	42.26

USB function core (USB), Ethernet IP core (Ethmac), MD5 full chip (MD5), and Reed Solomon decoder (reed_solomon) are from Opencore [12]. To introduce three random errors (Nr = 3), we use three types of errors, gate substitution, missing gate, and extra wire. Then we introduced 1 to 5 new random errors for each circuit and tried to correct the errors. To reduce the dependency on specific errors, we did this experiment 100 times for each case and computed the average repair rates as shown in Table 3. In the table, the number of cells and the total cell area for each circuit are shown in the 2nd and 3rd columns. Then the repair rates show the successful repair percentages among the 100 trials of random error repairs. The last row shows the average. On the average, when 11.64% of the spare cell area and 24.72% of the extended pins are added to the original circuits, 83.74% of the single errors (and 72.22% of the double errors) can be corrected by using two metal revision. Note that the design for the repair techniques can be applied only for error-prone blocks, so that the design overhead due to the spare cells and extended pins can be small at the chip level.

VI. CONCLUSIONS

In this research, we developed a new design repair method by using metal layer revision. When a “small” design error is found after mask production, partial metal revision can be an effective repair method which is very cheap and fast when compared to full mask revision. To define the appropriate repair block size, partitioning is used. To decide the amount of spare cells, partition granularity, and thus the number of extended pins, an automatic design for repair procedure is proposed, where the repair success rate and the cost can be controlled by three parameters, Kp, Nr, and Ke. A pseudo-cell technique is also proposed to use commercial placement and routing (P&R) tools during the metal layer revision. So far we have considered functional error repair techniques. Future work includes developing design repair techniques by considering the time delays on the critical paths.

ACKNOWLEDGMENTS

This research was supported in part by Samsung Electronics co.

REFERENCES

[1] M. Abramovici, P. Bradley, K. Dwarakanath, P. Levin, G. Memmi and D. Miller, "A Reconfigurable Design-for-Debug Infrastructure for SoCs," Proc. DAC ’06, 2006, p. 7.

[2] Collett International Research 2005.
[3] T. Jhaveri, I. Stobert, L. Liebmann, P. Karakatsanis, V. Rovner, A. Strojwas and L. Pileggi, “OPC Simplification and Mask Cost Reduction Using Regular Design Fabrics” SPIE Proceedings Vol. 7274, 13 March 2009
[4] Z. Or-Bach, "Paradigm Shift in ASIC Technology," eASIC Corp., Sep. 2005,
[5] K. Chang, I. L. Markov, and Valeria Bertacco " Reap What You Sow: Spare Cells for Post-Silicon Metal Fix" Proc. ISPD'08, April 13–16, 2008, pp.103 – 110
[6] K. Chang, I. Markov and V. Rertacco, "Automating Post-Silicon Debugging and Repair," Proc. ICCAD ’07, 2007, p. 91.
[7] S. Raman, and M. Lubyanitsky, "Cone Resynthesis ECO Methodology for Multi-Million Gate Designs", 22nd International Conference on VLSI Design, 2009, pp. 195 – 199
[8] K. Kim, B. Kang, D. Kim, S. Lee, J. Shin, and H. Shin, “Low-cost Design for Repair with Circuit Partitioning”, 15th Asia and South Pacific Design Automation Conference (ASP-DAC), 2010, pp. 259 – 261
[9] M.S. Abadir, J. Ferguson, and T.E. Kirkland, "Logic Design Verification via Test Generation", IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems, vol. 7, Issue 1, pp. 138-148, 1988.
[10] K. Y. Tong, V. Kheterpal, V. Rovner, L. Pileggi, and H. Schmit, "Regular Logic Fabrics for a Via Patterned Gate Array(VPGA)," Proc. of IEEE on Custom Integrated Circuits Conference, 2003, p. 53
[11] W. D. Cox, “Configuring structured ASIC Fabric Using Two Non-adjacent Via Layers”, US Patent 7,692,609 B2, Apr. 6, 2010
[12] Opencores. Available: http://opencores.org/

Sungchul Lee received the B.S. and M.S. degree in Electrical Engineering from Hanyang University, Ansan, Korea, in 2001 and 2003. He is currently working toward the ph.D. in Electronics and Communication Engineering at Hanyang University, Ansan, Korea. His research interests include design repair, multi-core design and low power design techniques.

Hyunchul Shin received the B.S. degree in Electronics Engineering from Seoul National University in 1978, and the M.S. degree in Electrical Engineering from the Korea Advanced Institute of Science and Technology in 1980. He received the Ph.D. degree in Electrical Engineering and Computer Sciences from the University of California at Berkeley in 1987. From 1980 to 1983, he was with the Department of Electronics Engineering at the Kumoh National Institute of Technology, Korea. In 1983, he received a Full bright scholarship. From 1987 to 1989, he was a member of the Technical Staff at AT&T Bell Laboratories, Murray Hill, NJ. Since 1989, he has been a professor with the School of Electrical and Computer Engineering of Hanyang University, Korea. His research interests include the design and synthesis of integrated circuits and systems for multimedia applications.