A LOGIC-AWARE LAYOUT METHODOLOGY TO ENHANCE THE NOISE IMMUNITY OF DOMINO CIRCUITS

Yonghee Im and Kaushik Roy

School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907, USA

ABSTRACT

The circuit performance is increasingly affected by signal integrity as cross-talk becomes more significant with scaling down of feature sizes. Many attempts have been made to improve noise immunity, but all require the sacrifice of speed as a tradeoff. In some circuits, P/G network is used as shielding wires to avoid cross-talk while maintaining the desired speed, but the use of the network is inherently restricted by electromigration, IR drop, $L\frac{di}{dt}$ noise, etc. We propose a novel methodology to enhance the noise immunity of domino circuits by reordering transistors as well as interconnects based on the functionality of the circuit. To the best of our knowledge, it is the first attempt to use the functionality of a circuit for the purpose of noise immunity enhancement. The methodology, named "Logic-Aware Layout Methodology" (LALM), is composed of a few sub-techniques and can be used to improve the signal integrity of domino circuits. Experimental results show that LALM is simple to apply yet useful to improve the noise immunity of domino circuits.

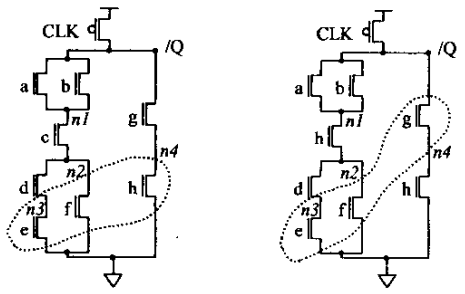
1. INTRODUCTION

In *gigascale integrated* (GSI) circuits, the signal integrity issue is expected to be more important as features scale down [1]. Increasing cross-talk, which is induced by unwanted coupling from neighboring signal wires to a certain node, is one of the most important factors for signal integrity. The signal integrity issue is more crucial in dynamic logic designs, because dynamic logic sacrifices noise immunity to achieve high performance. Conventionally, power-ground network (P/G network) has been used as shielding wires to achieve high noise immunity without speed degradation. Methods of obtaining optimal shield insertion and net ordering is well known as *Shield Insertion and Net Ordering* ($SINO$) problem. Lepak and He proposed several algorithms to solve the SINO problem [2, 3]. However, all these SINO algorithms are less practical because they use *sensitivity rate* which is not extracted from a real circuit but chosen rather arbitrarily. Further, all NO algorithms focus only on the reduction of capacitive coupling, not on the reduction of functional fault caused by cross-talk [4, 5, 6].

We propose a new methodology of reducing functional fault possibility by reconfiguring transistors as well as reordering nets based on the circuit functionality. The methodology is named *Logic-Aware Layout Methodology* ($LALM$). LALM is a practical algorithm, since it is based on no arbitrary assumptions. LALM is applicable to any dynamic logic designs. In particular, we have considered domino circuits in the paper.

The rest of the paper is organized as follows. In section 2, the principles and algorithms of LALM techniques are introduced along with pertinent experiments. Section 3 presents more comprehensive experimental results on circuits, applying all the techniques discussed in section 2.

This work was supported in part by Semiconductor Research Corporation



(a) circuit 1 (b) circuit 2

Figure 1: The combination of inputs to allow maximum charge sharing

2. LOGIC-AWARE LAYOUT METHODOLOGY

2.1. LALM/TO

The charge on the dynamic node leaks during evaluation mode in various forms [7]. Charge sharing is the most dominant form among them. In the circuit shown in Fig. 1(a), maximum charge sharing occurs when all inputs are high except for e, f, and h. The keeper transistor is sized in such a way that the logic operates properly under such a condition while still meeting the required propagation delay.

Let us assume that c is replaced by h in the circuit (Fig. 1(b)). Then, the above worst case condition is no longer valid, because nodes $n2$ and $n3$ are free from charge sharing. Instead, the maximum charge sharing occurs when inputs e, f, and g are low and all other inputs are high. The basic principle of *transistor ordering* in LALM ($LALM/TO$) is to explore transistor configurations to minimize the maximum charge sharing possible, while maintaining the functionality of the logic.

The search algorithm of LALM/TO is explained step by step taking the circuit shown in Fig. 1(a) as an example. In the first step, the circuit is transformed into a graph in which edges and vertices represent transistors and nodes, respectively (Fig. 2(a)). Then the graph is broken down into a set of $paths$ starting from the dynamic node ($/Q$) and ending at GND (Fig. 2(b)). In step 3, an edge is chosen, which appears on two or more paths. The edge is called *pivot edge*. In this example, edge a with two appearances is chosen as the first pivot edge (Fig. 2(c)). The next step

0-7803-7761-3/03/$17.00 ©2003 IEEE

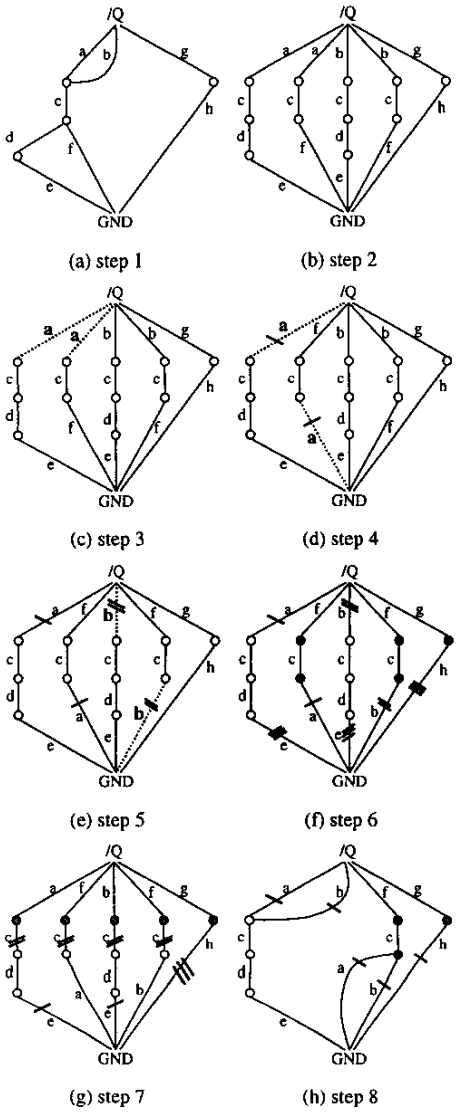


Figure 2: LALM/TO procedure

is to reorder the positions of the pivot edge. The positions are reordered in such a way that at least one of them is adjacent to GND (Fig. 2(d)). When the transistor represented by edge a is OFF, all nodes in the path $a-c-d-e$ will be free from charge sharing. This is the basic concept of LALM/TO to reduce charge sharing by reordering transistors. The process of reordering continues for all pivot edges (Fig. 2(e)). If all transistors adjacent to GND are OFF, the nodes denoted by shaded circle will have charge sharing (Fig. 2(f)). However, this may not be the worst case charge sharing, because pivot edges will keep some nodes free from charge sharing. On the other hand, the maximum charge sharing can be found in the same configuration, while keeping the transistors represented by all pivot edges ON this time. This is called *complementary* form of the configuration (Fig. 2(g)). Finally, the configuration can be simplified further by merging common edges, if available (Fig. 2(h)).

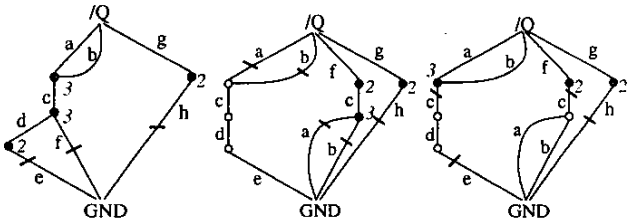


Figure 3: Comparison of charge sharing weight between the original configuration and LALM/TO configuration

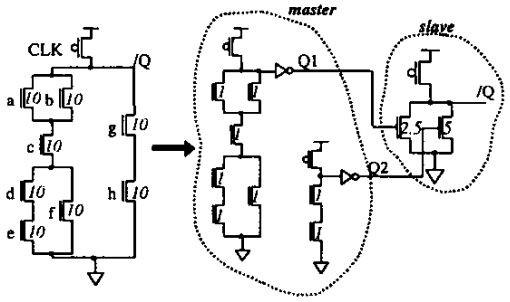


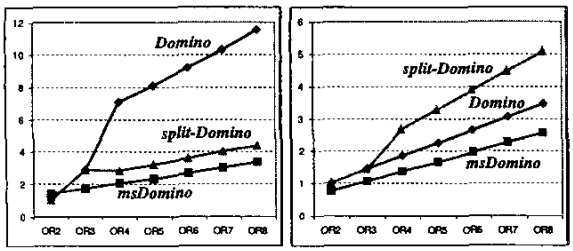
Figure 4: Breaking circuit 1 into the *master* and *slave* parts using LALM/msDomino

Charge sharing on a vertex can be represented with the degree of the vertex, since the parasitic capacitance on the node is proportional to the transistor width and all transistors are assumed to have the same width. The degree of the vertex is defined as the number of edges connecting to it. The sum of degrees on charge sharing vertices is named *weight* of the configuration. The weights of the final configuration and its complementary form are denoted by $W(a,b)$ and $W^C(a,b)$, respectively (Fig. 3). The worst case charge sharing of the reconfigured circuit is chosen as the larger weight between $W(a,b)$ and $W^C(a,b)$. The iteration continues for all possible combinations of pivot edges. We obtain minimum charge sharing weight, when a and b are used as pivot edges. 30% improvement in charge sharing weight is obtained by this process.

2.2. LALM/msDomino

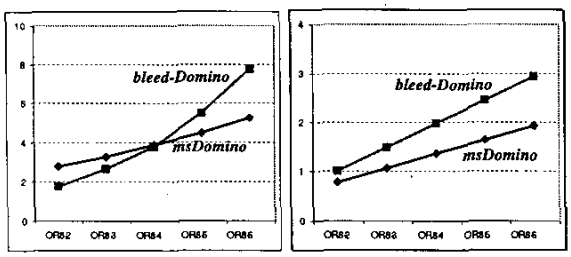
Better noise immunity can be achieved in LALM/TO without sacrificing performance, because the longest delay path from the dynamic node (i.e. $/Q$ in circuit 1) to GND remains the same as the original circuit. However, LALM/TO is not possible if no pivot edge exists. In this section, we propose an alternative method to minimize charge sharing when no pivot edge exists.

Realizing that charge sharing increases with the transistor size, noise immunity is better in PDN where smaller sized transistors are used. Using this concept, we propose a new method to break a domino circuit into two parts: (1) *master* and (2) *slave*. The *master* part is composed of small sized transistors in each cluster logic and the *slave* part consists of large sized transistors to create the final output. Such domino circuit is named *LALM/msDomino*. With LALM/msDomino, for example, circuit 1 can be implemented as shown in Fig. 4. The transistor width in the slave part is determined by the longest delay path of the master part. In addition to



(a) normalized PDP (b) normalized Area

Figure 5: Comparison of PDP and transistor area between domino, split-Domino, and LALM/msDomino



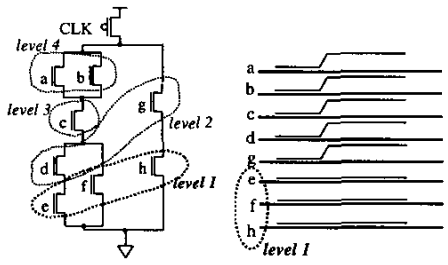
(a) normalized PDP (b) normalized Area

Figure 6: Comparisons of PDP and transistor area between bleed-Domino and LALM/msDomino

the small amount of charge sharing, if any, in the master part, the inverters between the master and slave parts help to improve the noise immunity of the circuit. However, longer delay is expected due to these inverters, or extra stage. This can be adjusted by using smaller keeper and smaller buffers.

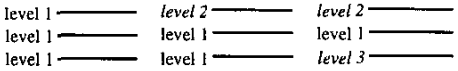
Similar to LALM/msDomino, a wide-OR gate (e.g. 8 input OR) can be implemented by splitting it into narrow gates (e.g. 2 input OR). Let us call it *split-Domino* in the paper. Split-Domino is also designed for higher noise immunity at the expense of delay. LALM/msDomino is compared with normal domino as well as with split-Domino in circuits having two through eight two-transistor clusters, denoted by *OR2*, *OR3*, ..., *OR8*. Spice simulations are performed on the circuits using a $0.18\ \mu m$ technology and performance metrics are measured at the same noise margin in the worst case in which all bottom transistors are coupled with rising neighbors. In this experiment, LALM/msDomino shows better results over split-Domino in terms of power consumption, noise immunity, area, and PDP (power-delay-product), while delay is almost the same. Fig. 5 compares normal domino, split-Domino, and LALM/msDomino in terms of normalized PDP and area among them.

Another method to minimize charge sharing is to attach small PMOS transistors to internal nodes of PDN. This PMOS transistor is called *bleeder* and the domino logic with bleeder is called *bleed-Domino*. The bleeder width is determined for a desired noise immunity while maintaining the delay constraint. LALM/msDomino is compared with bleed-Domino for circuits with eight-clustered gates with two to six transistors in each cluster, denoted by *OR8-2*, *OR8-3*, ..., *OR8-6*. Fig. 6 shows normalized PDP and area of LALM/msDomino and bleed-Domino circuits. As expected, bleed-Domino is worse in terms of PDP than LALM/msDomino as



(a) labeling transistors (b) net ordering

Figure 7: Application of LALM/NO into a single gate



(a) $M_i = 0$ (b) $M_i = 1$ (c) $M_i = 2$

Figure 8: Violation metric (M_i) in LALM/NO

the internal parasitic capacitance of PDN increases. LALM/msDomino also outperforms bleed-Domino in terms of area.

2.3. LALM/NO

Another way of improving the noise immunity of the domino circuit is to reorder nets. As mentioned earlier, the worst case in terms of charge sharing weight happens when all inputs to the transistors except for those with source terminal connected to GND are high. For convenience, transistors are labeled according to the distance from GND as shown in Fig. 7(a). Noise on *level 1* transistors induced by cross-talk worsens the noise immunity of the circuit by leaking charges. The basic principle of *net ordering* (*LALM/NO*) is to reorder the nets to reduce cross-talk on nets connected to *level 1* transistors. In circuit 1, the inputs of *level 1* transistors (nets e, f, and h) are separated from the other inputs in order to reduce cross-talk in the worst case as shown in Fig. 7(b). If shield insertion is available, the insertion of a shield wire between g and e improves noise immunity further.

However, applying LALM/NO in circuits consisting of multiple gates is not simple. Let us define *violation metric* (M_i) for the cross-talk susceptibility of the *level 1* nets of a logic gate. The cross-talk susceptibility of the circuit (M_T) is the summation of the violation metric of each logic gate. Violation metric of a net is allowed to have three different values as shown in Fig. 8: '0' when no neighbors of the *level 1* nets belong to other levels, '1' when only one neighbor is in other levels, and '2' when both neighbors are in other levels.

LALM/NO searches for an optimal net order leading to the minimum M_T with a greedy algorithm. The search algorithm shuffles nets so that the first gate has the minimum metric M_1. Then, it does the same thing for the second gate while maintaining or decreasing M_1. The algorithm continues to do the same thing for all the gates. This greedy search always ends up with an optimal solution.

With a set of gates shown in Fig. 9, let us search for both the best order and the worst order of a given set of nets, a through h, in terms of violation metric. The worst net order can be obtained in the opposite way, i.e. shuffling nets to obtain a maximum M_T.

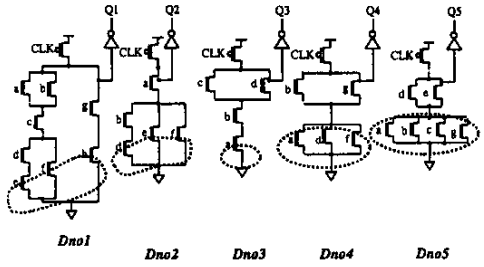


Figure 9: A set of gates used in LALM/NO

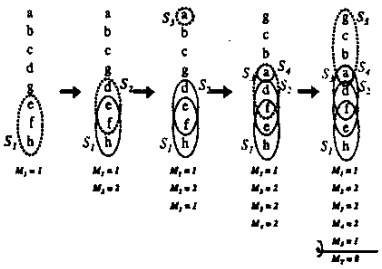


Figure 10: Application of LALM/NO into multi-gates

Fig. 10 illustrates how to search for the best net order. S_i is the set of *level 1* nets of the logic gate i. The elements of S_i are shown in a dotted circle in Fig. 9. M_i indicates the violation metric of gate i and M_T is the violation metric of the set of gates. Consider the net order shown in Fig. 7(b). In gate 1, nets e, f, and h are in level 1. There is no adjacent net to f and h which is in other levels. Therefore, the violation metrics of f and h are '0'. However, net g which is in level 2 is adjacent to e. Therefore, the violation metric of e is '1'. Hence, the violation metric of gate 1 (M_1) is '1'. Similarly, nets d, e, and f are in level 1 of gate 2 and net order is shown in the second column of Fig. 10. Here, net g is adjacent to d. Therefore, d will have the violation metric equal to '1'. Similarly, net h is adjacent to f and thus f will also have the violation metric of '1'. The violation metrics of all other gates are evaluated in a similar way.

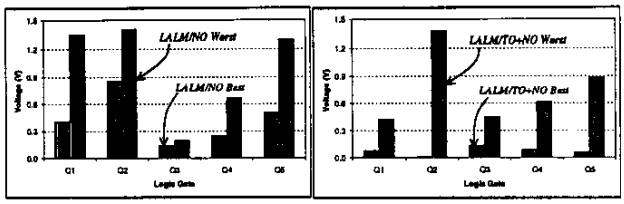
Applying a test vector set, with which maximum charge sharing occurs in each gate, the output voltage level of each gate is measured and plotted for both the best and the worst net order as shown in Fig. 11(a). The output voltage level of each gate should be closer to $0V$ without error. However, the worst case shows larger voltage level than the best case, even generating functional fault in some cases ($Q1$, $Q2$, and $Q5$), i.e. $V_{out} > V_{DD}/2$ (=1.5/2).

3. EXPERIMENTS

In this section, we show the effectiveness of LALM/TO and LALM/msDomino when combined with LALM/NO.

3.1. LALM/TO + NO

The set of gates shown in Fig. 9 is reconfigured into the worst and the best case w.r.t. the charge sharing weight using LALM/TO. LALM/NO algorithm is applied to search for the worst net order for the worst LALM/TO as well as the best net order for the best



(a) LALM/NO (b) LALM/NO + LALM/TO

Figure 11: Comparison of output voltage of gates in multi-gates

LALM/TO. The similar search steps are performed and the resulting output voltage level of each gate is plotted in Fig. 11(b). As expected, the combination of LALM/TO and LALM/NO improves noise immunity further.

3.2. LALM/msDomino + NO

Similarly, LALM/msDomino combined with LALM/NO is compared with LALM/msDomino. The logic gates used in the comparison between LALM/msDomino and bleed-Domino are also chosen in this experiment. The same test vectors are used to measure power consumption and delay. PDP is compared between LALM/msDomino and LALM/msDomino+NO. PDP of LALM/msDomino+NO is lowered by 23 % from that of LALM/msDomino, because the amount of leakage current caused by cross-talk sufficiently decreases due to net reordering.

4. CONCLUSIONS

We proposed a novel methodology, named *Logic-Aware Layout Methodology* ($LALM$), to improve noise immunity based on the functionality of PDN of domino circuits. LALM is categorized into transistor ordering (TO) and net ordering (NO). LALM/TO and LALM/msDomino are suggested as TO method. LALM/TO methods are compared with conventional domino circuits and other precharge-evaluate circuits. Moreover, LALM/NO is suggested to reduce cross-talk causing functional fault. Spice simulation results show that both LALM/TO and LALM/NO improve the noise immunity. We also observed that, using both methods simultaneously, one can improve the noise immunity of the circuit further.

5. REFERENCES

[1] J. D. Meindl, "Gigascale integration: Is the sky the limit?," *IEEE Circuits and Devices*, vol. 12, no. 6, pp. 19–24, 32, 1996.

[2] L. He and K. M. Lepak, "Simultaneous shield insertion and net ordering for capacitive and inductive coupling minimization," *IEEE/ACM International Symposium on Physical Design*, pp. 56–61, 2000.

[3] K. M. Lepak, I. Luwandi, and L. He, "Simultaneous shield insertion and net ordering under explicit rlc noise constraint," *Design Automation Conference*, pp. 199–2022, 2001.

[4] T. Gao and C. L. Liu, "Minimum crosstalk switchbox routing," *IEEE/ACM International Conference on Computer-Aided Design*, pp. 610–615, 1994.

[5] T. Xue and E. S. Khu, "Post global routing crosstalk synthesis," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 1997.

[6] J. Yim and C. Kyung, "Reducing cross-coupling among interconnect wires in deep-submicron datapath design," *Design Automation Conference*, 1999.

[7] J. M. Rabaey, *Digital Integrated Circuits : A Design Perspective*, Prentice-Hall, 1996.