A Knowledge Based Design Process Addressing the Antenna Effect and Cell Placement

Peter H. Chen and Martin Ngu

2833 ZettaCom, Inc.
Junction Avenue,
San Jose, CA 95134

Pchen@zettacom.com
mngu@zettacom.com

Larry Roffelsen

Teralogic, Inc
1240 Villa Street
Mountain View, CA 94041

lroffelsen@teralogic-inc.com

ABSTRACT

This paper proposes a simplified design process containing solutions to both the antenna effect problem and increasing the efficiency of cell placement for million-gates chip layout

1.0 Introduction

Working within the ASIC industry, time to market is very important. With Very Deep Sub-Micron (VDSM) processes, issues with timing enclosure, cell placement efficiency, the antenna effect, signal integrity, etc. have impacted our ability to meet market requirements in a timely manner. By focusing on the design process, two of the major issues, the antenna problem and cell placement efficiency, can be addressed. I propose using Knowledge Based Priority Placement (KBPP), which utilizes design knowledge to simplify those two issues.

2.0 Antenna Effect

Problem

The antenna problem has existed in the semiconductor industry for more than a decade. In the production of the modern VLSI semiconductor, etching plasma is used to deposit or remove material on wafers. The etching plasma causes destructive charges to be built up on the wafer. This problem is more critical in VDSM VLSI design since thinner oxides are even more susceptible to damage. During wafer manufacturing, aluminum or polysilicon wires in the plasma ambient that are not covered with dielectric collect charges and serve as "antenna." When the potential becomes high enough, it will discharge through the gate dioxide and cause cell damage. Damage can include electrical stress that decreases the breakdown voltage of the gate oxide, resulting in a decreased yield rate during wafer manufacturing. It has been hypothesized that UV light bombardment on the etching plasma during the manufacturing process increases the antenna effect; however, I have found that it increases the effectiveness of protection diodes since they are the miniature solar cells and they conduct the charge instead of the gate dioxide. [3, 5, 12, 14, 15]

Possible Solutions

To prevent this, we can dig a drainage trench, reduce the antenna lengths, insert jumpers in the post-layout phase, or connect protection diodes as close to the input ports as possible. The first two options are not feasible as they necessitate changing the design of a completed chip during the manufacturing process. Among these solutions, only post-layout jumper insertion and protection diodes can be used in the physical design period. However, even smart routers are not able to insert jumpers because chips are so dense.

Jumpers also cause more time delay than diodes and cause timing enclosure problems. The following calculation shows the timing comparison between jumper and diode insertion approaches. In jumper insertion, each jumper needs at least two vias. In 0.35 μm technology, via resistance is around 10 Ω. In 0.25 μm technology or below (such as 0.18 μm), via resistance is around 100 Ω or more. The bigger the resistance, the worse the timing delay is. This phenomenon is more severe in deeper sub-micron technology.

$$R' = R_s \times \frac{L}{W} \cong 60m\Omega \cong 3-10\Omega \qquad (1)$$

Where:

R_s: The sheet resistance of metal.

R': The resistance of metal with length L.

If we use diode insertion, the diode capacitance is the factor that affects the timing delay. The

following equation estimates typical diode capacitance.

$$C' = C_{ox} = \frac{\varepsilon_{ox}}{t_{ox}} = \frac{35.5\mu F / cm}{0.4\times10^{-5} cm}$$

$$= 0.86\, fF / \mu m^2 \qquad (2)$$

Where:

C : Capacitance of the original wire

C' and C_{ox} : capacitance of the inserted diode

ε_{ox} : Permittivity of diode

t_{ox} : Thickness of the diode

If we add one jumper (with two vias) and one protection diode for a wire 350 μm long, we have the following estimate timed delay.

$$\tau = (R + R')\times(C + C')$$

$$= R\times C\times(1 + R'/R)\times(1 + C'/C) \qquad (3)$$

Assume a wire with 60 Ω and capacitance $1pF$. If we compare dropping 2 vias at 3 Ω (total 6 Ω) versus dropping a protection diode at $0.86\, fF$ from equation (3), we can see that the delay caused by the protection diode $(C'/C \cong 0.86\, fF / 1pF \cong 10^{-3})$ is much less than that caused by the jumper insertion [or router option] $(R'/R \cong 6/60 \cong 0.1)$. Therefore, the timing delay caused by dropping the diode is negligible.

Simulation Results

Figures 1 and 2 simulate the inverter with a protection diode during the manufacturing process. The protection diode size also affects the cumulative current. Assume a current pulse is generated from the etching plasma into the input gate. The input gate is protected by a protection diode with area $1\times1\mu m^2$. The plasma current source (Is) is set to the following conditions: current amplitude 10 mA, period 20 nsec, pulse width 5 nsec, 1 nsec in both the rising and falling edge, and initial delay 2 nsec. Resistors R1, R2, and R3, are set to 5 Ω. Capacitances C1, C2, and C3 are set to 0.75 pF. Both N-MOS and P-MOS are tied to ground. Voltages are measured and displayed at node NI and the input port (node N3). Since the protection diode provides drainage, the plasma current at the input gate is kept to zero (Figure 2).

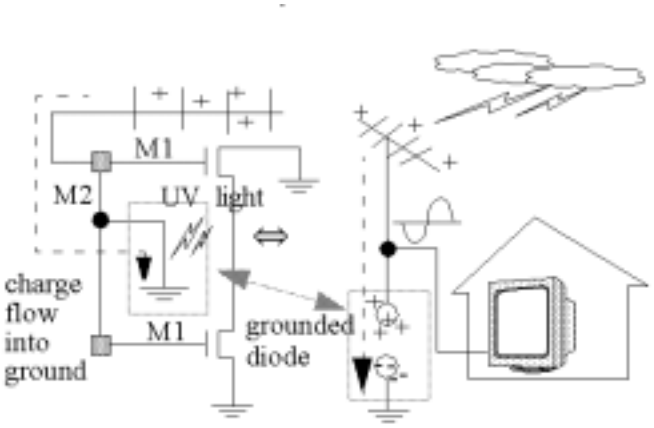

Fig. 1. Analogy of protection diode fixing antenna problem during manufacturing.

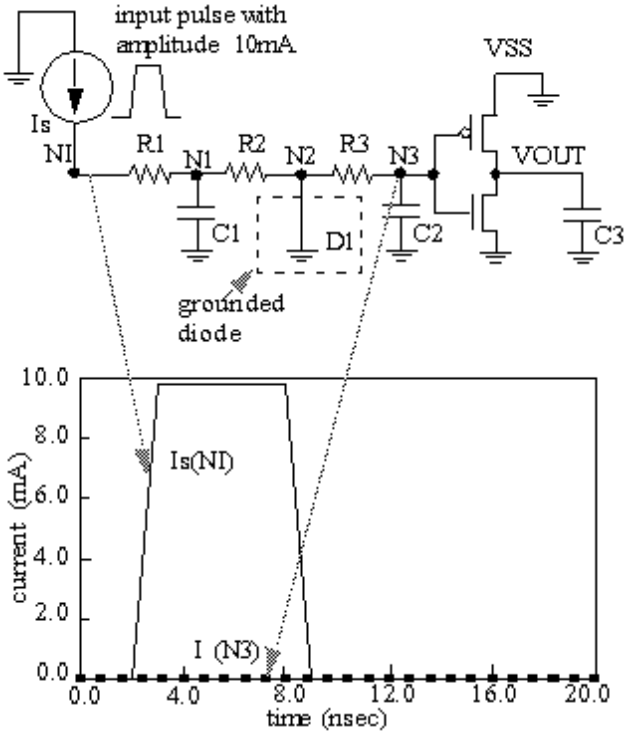

Fig. 2. SPICE simulation: input port is grounded by the protection diode.

Figure 3 illustrates the normal operation of a circuit when voltage is applied to the chip. The SPICE model during circuit operations (in Figure 4) shows that the diode does not interfere with the integrity of the signal when a pulse is input through the input gate. The protection diode has an area of $1\times1\mu m^2$. The input current source (Is) is set to the following conditions: current amplitude 10 mA, period 20 nsec, pulse width 5 nsec, 1 nsec in both the rising and falling edge, and initial delay 2 nsec. Resistors R1, R2, and R3, are set to 5 Ω. Capacitances C1, C2, and C3 are set to 0.75 pF. Both N-MOS and P-MOS are tied to ground. Voltages are measured and displayed at node (I) and the input port (node N3).

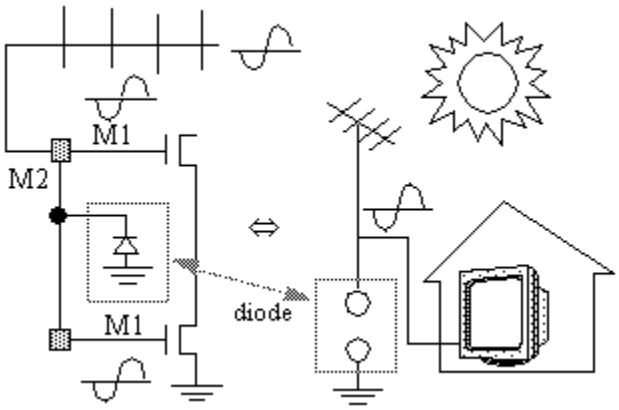

Fig. 3. Analogy of normal circuit operation.

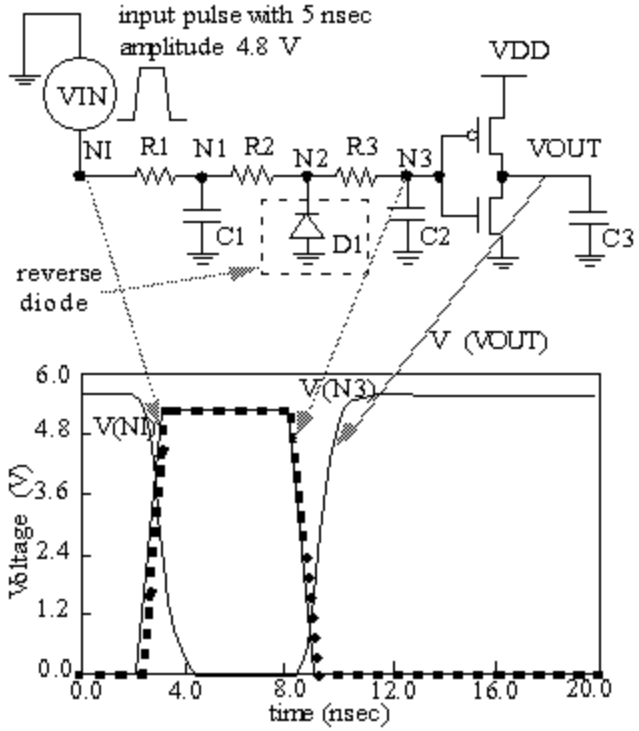

Fig. 4. SPICE simulation: Input port is protected by diode during the normal operation.

In Figure 4, the shift between the solid lineV (NI) and the dotted line V (N3) is caused by the capacitance and resistance between them. The area of the diode (e.g., increasing the diode size from $1\mu m^2$ to $3\times10^{14}\,\mu m^2$) has a negligible effect. The leakage currents introduced by the protection diode are discussed in previous papers [3, 5].

Antenna Ratio: A Typical Example

Note that the antenna rule violation definition is foundry and process specific. Using a typical 0.25 μm (5 metal) process design rules as an example, the antenna ratio is defined using the total

perimeter area of metal wire as the numerator and the gate area as the denominator.

$$\text{Antenna Ratio} = \frac{\left[2 \times (L1 + W1) \times t\right]}{W2 \times L2} \qquad (4)$$

Where:

$L1$: floating metal length connected to gate
$W1$: floating metal width connected to gate
t : metal thickness
$W2$: connected transistor channel width
$L2$: connected transistor channel length

The antenna ratio for a typical 0.25 contact (CO) and Via1 -Via4 is:

$$\text{Antenna Ratio} = \frac{\text{Total Contact (Via) Area}}{W2 \times L2} \qquad (5)$$

The thickness of M1-M4 is $5700\,\text{Å}$. The thickness of M5 is $9900\,\text{Å}$. The following diagram shows the dimensions of the above equation.

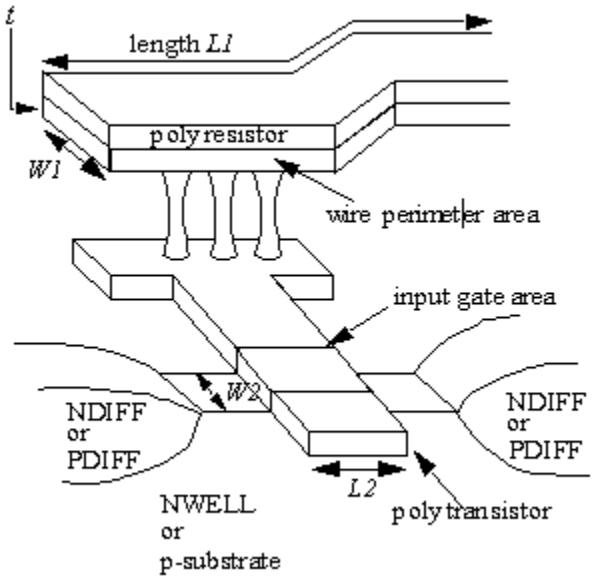

Fig. 5. Dimension definition of a typical 0.25 μm process.

Protection Diode Structure

Protection diodes are categorized as N-Diode with N-IMP (N+ implant) or P-Diode with P-IMP (P+ implant) in this paper. Fig. 6 shows the top view of the protection diode structure.

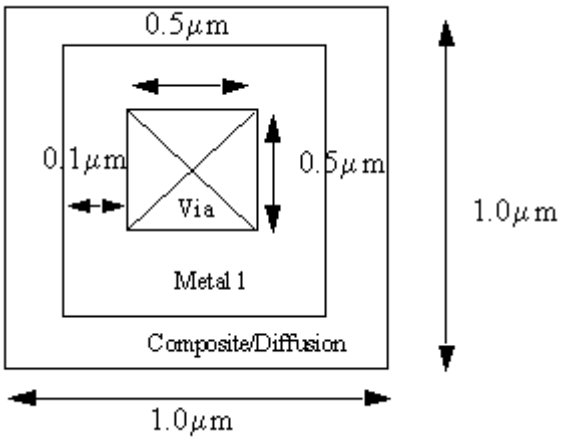

Fig. 6. Protection diode top view.

Fig. 7 and Fig. 8 show the side view of a P-Diode for the P-MOS process and an N-Diode for the N-MOS process, respectively. N-IMP, P-IMP, and DIFF are the fundamental mask layers. PDIFF and NDIFF are derived from these fundamental mask layers. Both types of protection diode can be dynamically dropped after the placement is done. Since the protection diode is very small $1\times1\mu m^2$ compared to the cell size, it is not difficult to find empty area (N-MOS or P-MOS) to drop the protection diode or readjust a little bit of the cell location to get the space after the final placement.

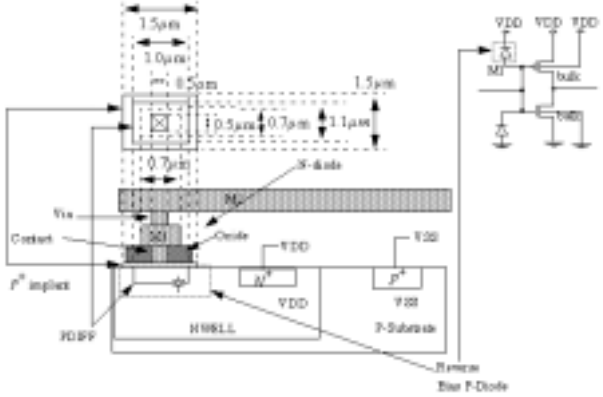

Fig. 7. P-Diode structure for P-MOS process

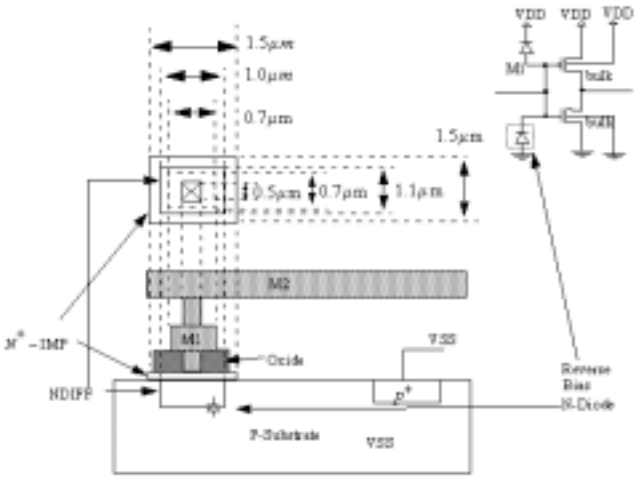

Fig. 8. N-Diode structure for N-MOS process

Leakage

Protection diode solutions are not without disadvantages, however. Currently, cell library vendors embed 1-3 protection diode(s) on each input pin. Because each standard cell contains multiple input pins, million gates chip designs would then contain several million protection diodes. According to our experience the majority of those diodes are not used. In addition to increasing manufacturing costs, the capacitance of each diode produces a very small leakage current, decreasing chip performance and leading to timing degradation. In fact, we experienced timing degradation up to 60% or worse, i.e., a desired 100 MHz chip may operate at 33.3 MHz when millions of diodes are employed. In fact, we have reduced the number of protection diodes needed from millions to hundreds by dropping them as needed at locations of antenna violations. Therefore, the post-layout dynamic protection diode dropping approach seems to increases efficiency in both performance and cost. However, due to the post-layout nature, even the reduced number of diodes is difficult to place on a congested design resulting in a failure to fix the antenna problem after the routing is done.

Proposed Solution

The antenna effect problem should be solved after final cell placement and before routing. Protection diodes are very small (about 1-micron by 1-micron) compared to the standard cell size. They are easily inserted between the rows, cells, or anywhere in the design. At this phase, we can adjust cell location and avoid congestion or rerouting (which is extremely time consuming and difficult in the post layout phase). It is possible to reasonably predict where antenna problems will arise. By enclosing each input and output pin within a bounding box and calculating the half-perimeter [4, 10], we can pinpoint values that will cause antenna violations. Note that in situations of multiple fanout (one output pin drives several input pins) the summation of multiple half perimeters will be required. Even if this method of estimating antenna violations is not 100% accurate, congestion will be so much reduced that any violations discovered in the post-layout phase will be corrected much more easily.

3.0 Cell Placement

Problem

In order to avoid wasted space in the hierarchical design, our hierarchical design blocks are flattened into the top level which contains 1.2 million logic gates. Only the third party library blocks such as DAC, PLL, and memory blocks are still kept. Commercial tools using generic approaches (such as, quadratic programming, genetic algorithm, genetic evolution, or simulated annealing) have been unable to fully optimize efficiency with these approaches. Quadratic programming without constraints results in overlapped cells. [10, 17] Constrained quadratic programming requires linearization of the constraints by taking the Taylor expansion of the constraints. For blocks with more than 25 modules, it is not practical to solve linear equations with these linearized constraints for millions of cell modules. In addition, the constrained quadratic programming can be used to solve rectangular modules, but not the rectilinear modules common to ASIC design. [4]

Cell placement currently can take from several days to several months since cell placement and routing are NP-problems and the time to solve them grows in an exponential manner. For the same reason, the tools exhaust memory resources resulting in core dump problems. Both these issues negatively affect time to market. Mathematicians continue to struggle for a solution to the NP-problem. This is not the approach I choose to take. Instead, I attempt to use a heuristic approach to simplify the existing problem domain. In other words, it is possible to use knowledge of design and design constraints to simplify the process without solving the NP-problem. [1, 2, 6, 8, 10, 11, 13, 16, 17]

Proposed Solution

This paper proposes a simplified model that uses IO-constraints based on the given package type. This method is not suitable for initial prototyping or other situation where the IO position is unknown. Using the circuit levelization offset and the algebraic median of the associated IO position, the initial cell placement is obtained. Circuit levelization determines: 1) the possible row numbers for each cell; 2) the maximum number of siblings for each cell; and 3) the maximum level for each net by traversing the netlist connectivity from IO input to IO output. The final placement is prioritized by timing constraints and is determined through combinatorial annealing. The annealing area is bounded by multiple fanout, i.e., determining the column position inside each row based on the timing cost function; and by multiple circuit levels, i.e. determining the row number inside each column based on the timing cost function.

Combinatorial Annealing Model

Assume the standard cells (or gates) are positioned at the center of each cell. Knowledge Based (Restricted Area) Annealing is used to find the best sub-optimal solution for cell location within the rows, for a cell belonging to a single circuit level, or within an area for a cell belonging to multiple circuit levels. Some other annealing techniques can be combined with restricted area annealing.

The total wire length between gates,

$$= \frac{1}{2}\left[\sum_{i=1}^{M}\sum_{j=1}^{N_M}\left[c_{ij}\cdot(x_i - x_j)^2 + c_{ij}\cdot(y_i - y_j)^2\right]\right] \quad (6)$$

Total wire length between gates and IOs (boundary constraints)

$$= \sum_{i=1}^{M_{io}}\sum_{k=1}^{N_{io}}\left[c_{ik}\cdot(x_i - x_k)^2 + c_{ik}\cdot(y_i - y_k)^2\right] \quad (7)$$

Where $c_{ij} = 1$ if $i \neq j, c_{ij} = 0$ if $i = j$

Where M is the number of gates, N_M is the number of nets in each gate, N_{io} is the number of gates connected to the IOs, and c_{ij} is the weighting of the net.

The timing constraints for a critical net from source register (*s*) to destination register (*d*) can be estimated by the half-perimeter during the placement period:

$$t_{\min} \leq \sum_{i=1}^{N_{rc}} R_i C_i + \sum_{i=1}^{N_g} \text{gate}_i + \text{ant\_del} \leq t_{\max} \quad (8)$$

In equation (8), the first term is the *RC* delay of this critical path. The delay calculator can be either Elmore or Asymptotic Waveform Evaluator (AWE). [4] The capacitances are calculated from the estimated metal distribution of each layer with the intrinsic, coupling, and fringe capacitances. The second term is the gate delay in the selected critical path. The term "ant_del" represents the delay caused by the protection diode insertion. The delay data can be roughly estimated by the half-perimeter method or calculated from pre-routing. Protection diode delay can be estimated by equation (5) with $R' = 0$ and $C' = 10^{-3}$ if only one protection diode is dropped in the specified critical net.

Equations (6), (7), and (8) can be rearranged with parameters defined as above into:

Minimize:

$$\frac{1}{2}\left[\sum_{i=1}^{M}\sum_{j=1}^{N_M}\left[c_{ij}\cdot(x_i - x_j)^2 + c_{ij}\cdot(y_i - y_j)^2\right]\right]$$

$$+ \sum_{i=1}^{M_{io}}\sum_{k=1}^{N_{io}}\left[c_{ik}\cdot(x_i - x_k)^2 + c_{ik}\cdot(y_i - y_k)^2\right] \quad (9)$$

Subject to:

$$t_{\min} \leq \sum_{i=1}^{N_{rc}} R_i C_i + \sum_{i=1}^{N_g} Gate_i + ant\_del \leq t_{\max} \quad (10)$$

Equation (9) and (10) are used to anneal the final cell placement.

KBPP

Figures 9 and 10 show the traditional (generic) design flow versus the proposed Knowledge Based Priority Placement (KBPP) design flow. In the generic design flow, timing constraints and IO constraints are optional. In KBPP, the timing constraint, IO constraints, and dynamic protection diode insertions are required. KBPP uses the constraints as pre-knowledge to determine the cell's locations to replace the traditional quadratic programming approach, which is

time consuming and results in overlapped cell locations.

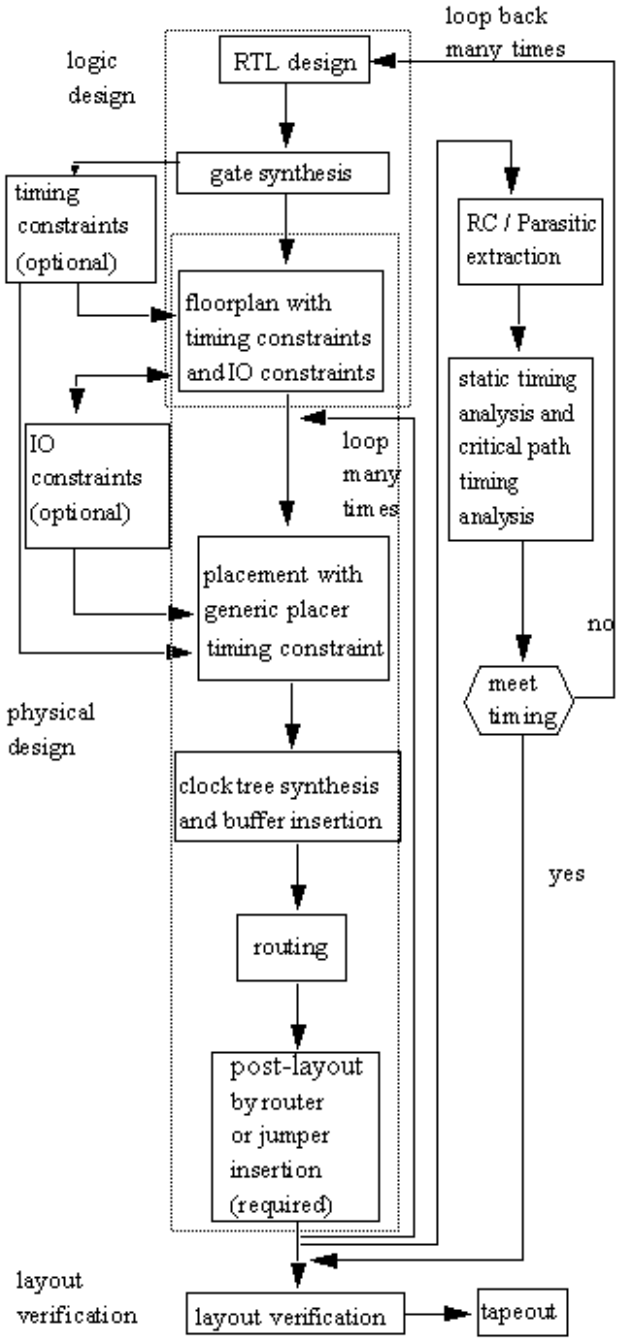

Fig. 9. Traditional flow.

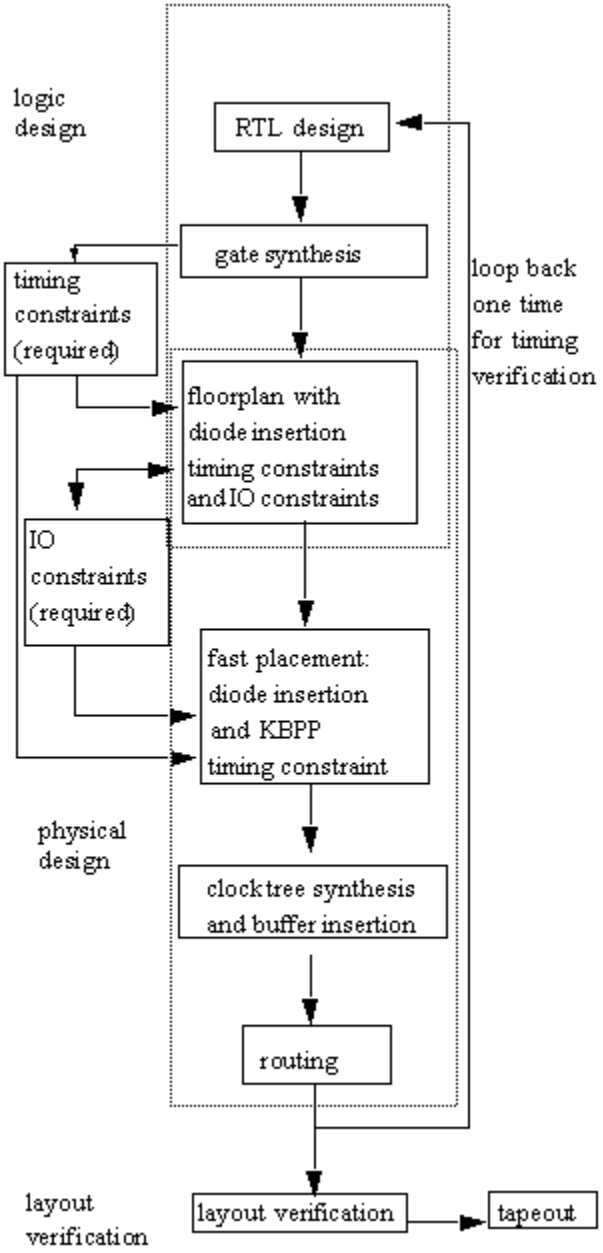

Fig. 10. KBPP flow.

Floorplanning and Timing Constraint Preparation

1) Carefully design the top-level pin orientations and location of large blocks, such as memory, DAC, PLL, and datapaths. Blocks should be placed near the top-level boundary and their pins placed along the sub-block boundaries that do not overlap the top-level boundary. Top-level pins should be placed along the boundary away from the overlapping area, also. Determination of IO locations and directions is the most critical step at the beginning of chip layout. It is necessary to perform several IO-layouts and placements to examine the routing congestion in both rows and columns. Once the locations are determined, revised chips should maintain the IO-locations.

2) For prototype chip designs, carefully constrain the IO position with the designer and with floorplanning tools based on connectivity. After the IO position is set on the prototype, subsequent revisions do not need to go through this step.

3) Generate the timing constraint for the critical net and check whether or not it is reasonable. The timing constraint for all critical paths should be prioritized, e.g., from one register (dstw) to the RF register is 8 $n\text{sec}$, tstd register to HIF block is 7.8 $n\text{sec}$, etc. Those critical paths should be prioritized (by logic design) and placed within a bounded region (in physical design).

4) Add protection diodes to nets with antenna violations (Fig. 11). The typical size for the protection diode (or a tap) is about $1\mu m^2$ which is very small compared to the typical standard cell size (e.g., JK-Flip Flop size $35\mu m \times 10\mu m$). The protection diode (either n-protection diode or p-protection diode) should be attached as close to the input port as possible. The antenna violation wire can be determined by traversing the placement cell location with the netlist associated with them. The protection diodes should never be dropped on power and ground rails. The dropped protection diode and JK-Flip Flop are shown in Fig. 12.

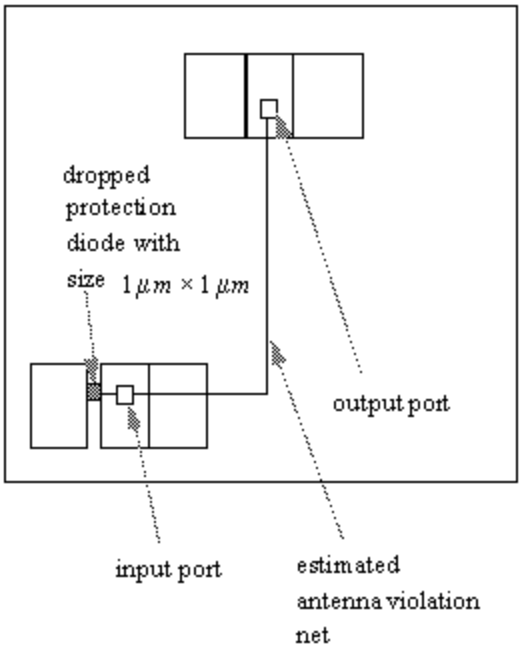

Fig. 11. Dropping the protection diode during the placement period

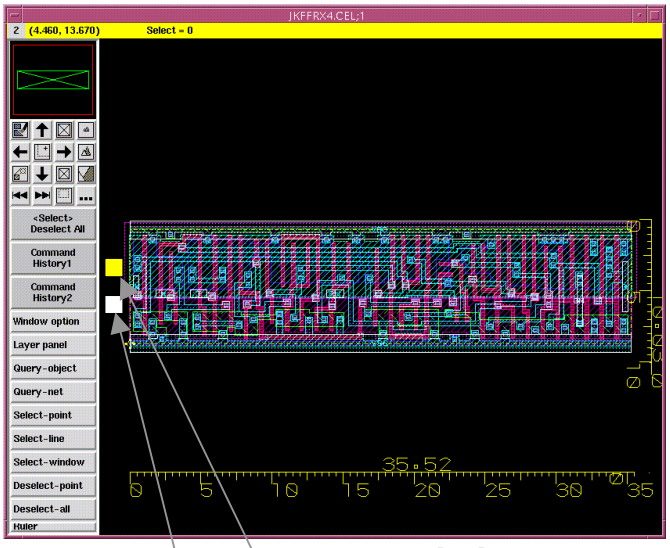

Fig.12. Dynamic drop n-protection diode or p-protection diode during the placement.

Cell Levelization

In KBPP, the cells are arranged by Horizontal Levelization and Vertical Levelization (such as VL-1, VL-2, VH-1, VH-2, etc.) while traversing the netlist as shown in Fig. 13. Due to the different paths traversing, each cell can have different of circuit levels. Those circuit levels are stored in a linked lists. If the cell is a single circuit level, row number for this cell is fixed. If the

cell contains multiple circuit level, the final cell placement performs the restricted area refinement based on multiple circuit levels.

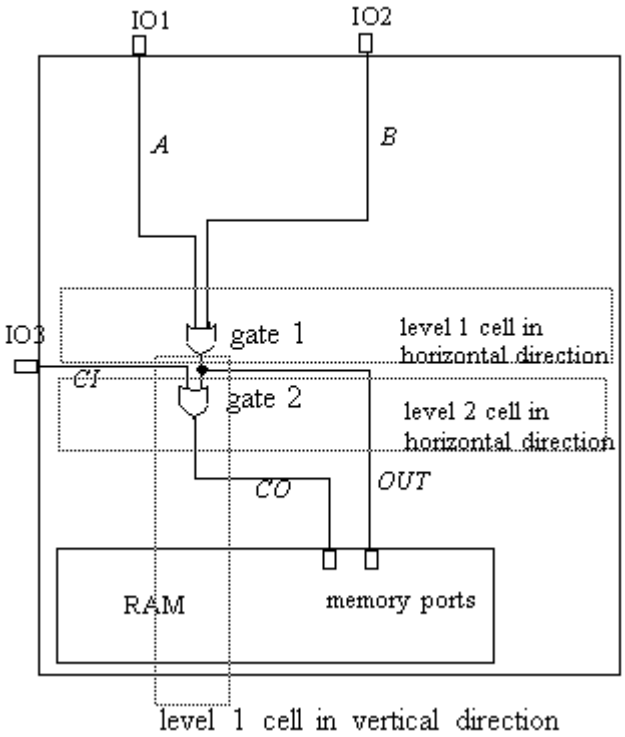

Fig. 13. Cell levelization.

Initial Placement

Assume the IO and macros location / directions are properly floorplanned. The locations of initial cell positions are obtained by simple algebraic median plus the level offset of the cell. If the maximum number of rows for any cell associated with a specific net exceeds the given number of rows, multiple columns or rows are required to place the cells. The multiple columns for a specified net can be determined by the following equation:

$$Mul\_col = round(\frac{Circuit\_level}{Max\_col}) + 1 \qquad (11)$$

Where:
Mul_col: is the column number will be placement for the specified net.
Circuit_level: is the number of the circuit level after the circuit levelization.
Max_col: is the maximum column specified by the user.

Cell congestion along one row can be treated in the same way.

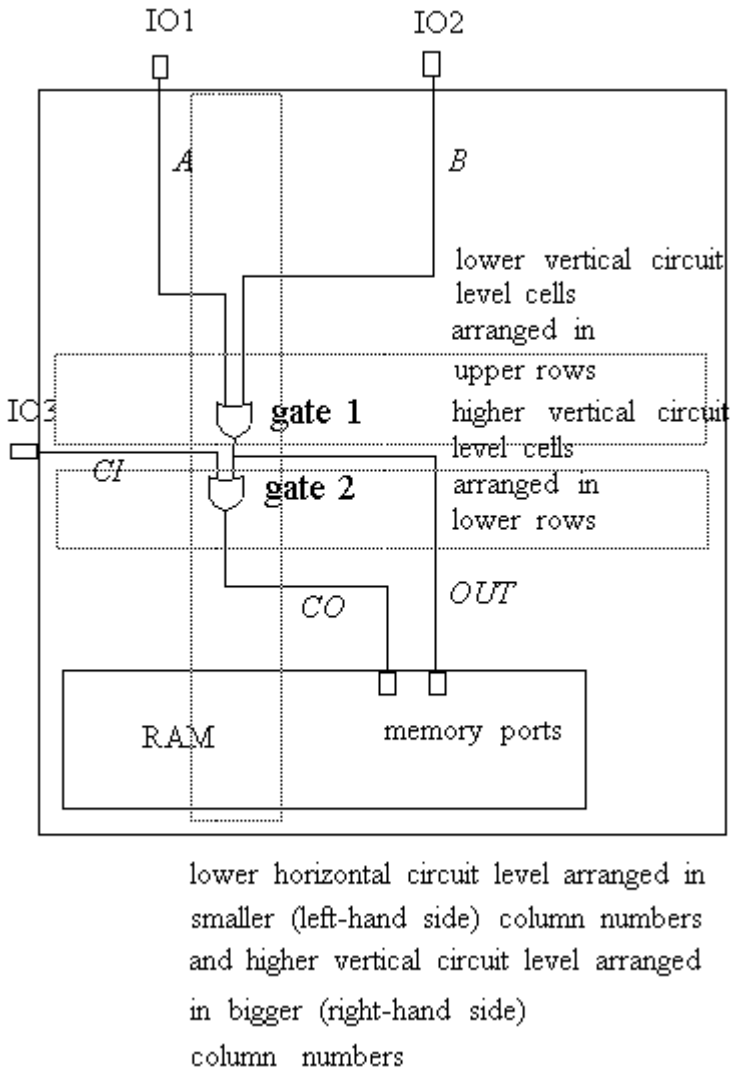

Fig. 14. Initial placement with the cell levelization.

Analyze the Placement with Histogram

Analyze the vertical congestion, horizontal congestion, vertical cell area occupancy, and horizontal cell area occupancy. This data provides direction for iterative placement improvement. This step is mainly useful for refining cases of highly asymmetrical or highly fanned out circuitry.

Perform the Post-Layout Antenna Problem Fix

There should be no antenna problem since protection diodes are added during the placement period. If antennae exist, dynamically drop protection diodes to fix the antenna net. Some extension wire can be added to provide more space to add protection diodes. A smart router, which re-route the antenna violation net, can also be tried. Start from the top of the input port of the topmost layer and route down to the output port.

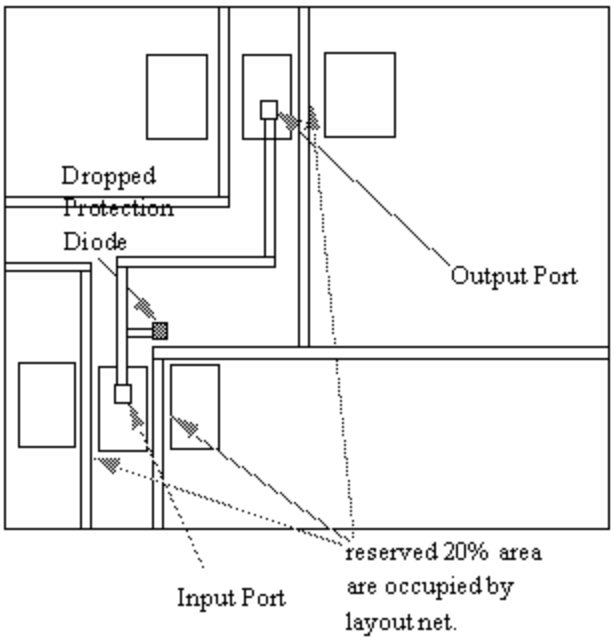

Fig. 15. Dropping protection diode during post-layout using extension wire due to congestion.

KBPP Algorithm

The major KBPP algorithm is summarized as follows:

1. Prioritize the netlist by timing constraints.
2. Traverse each cell by the netlist. Find all the associated IO cells for each net and levelize the cell both vertically and horizontally.
3. Traverse netlist and find cell locations by taking the median of the cell's IO locations and the offset of the circuit levelization.
4. Perform the final placement.
5. Map vertical and horizontal congestion and adjust, arrange, or fold the cell position by modulating the row number. This step is for asymmetric layouts, such as datapath blocks or high fanout circuitry.
6. Traverse net and find antenna violation locations. Estimate wire length using the half-perimeter of the bounding box. [4] Place a protection diode at the input port of cells associated with the antenna violations.
7. Perform clock synthesis and routing.

Result Comparison

The following table shows the comparison between the KBPP with post-layout antenna fixing and quadratic placer.

	Post-Layout Jumper Insertion with Smart Router	Post-Layout Dynamic Protection Diode Insertion	KBPP
Circuit Placement Time	5 hours (Sun Sparc with 8 CPUs)	4-6 hours (Sun Sparc with 8 CPUs)	20 minutes on a PC
Antenna Violations	Fixed 156 violations. 2 remaining.	Fixed 160 with 5 remain. Router has difficulty to find the re-route space.	All fixed.
Chip Area	10% reserved for congestion.	5% reserved for congestion.	0% extra area reserved for antenna fixing
Timing Requirement	Not Met	Met. Best timing delay in critical net 6.2 nsec.	Met

Table 1. Result comparison.

Chip Example

This work uses two chip designs to illustrate the differences of the generic approach and the KBPP approach. The first design is the hierarchical design containing one DAC block, one PLL, and about 10 hierarchical glue logic blocks. Each level of modules contains around 100K-300K cells. The generic solution is able to handle it on each level of hierarchy properly. The second design is flattened containing 2-DAC blocks, 3-PLLs, and approximately 250K more glue logics than the first example. The number of gates on the top level is about 1.2 million. Flattening the design allow the chip to hold 25% more circuitry than with the hierarchical design. However, the generic solution is not able to handle cell placement. The resulting cell placement could not resolve timing violations. [4] In addition, the process took too long.

The following diagrams show the TeraLogic designs used as test cases. A Sun-Sparc™ workstation with 8 CPUs running in round robin fashion at 400 MHz with 14 GB of main memory per thread was used to perform the first two approaches to cell placement and antenna fixing. The KBPP placement was performed on a PC and the results were transferred in Physical Design Exchange Format (PDEF) or Top Design Format (TDF) to the workstation. [4, 7, 8] In the hierarchical chip, since the number of cells on the top level is within generic can handle, both the generic and KBPP approaches working fine. For the flatten type of chip with about 1.2 million cells (standard cells, filler cells, IOs, macros, pad cells, IO filler cells) on the top, the generic approach is not able to handle it properly. KBPP or other in house tool using the simple cut approach can meet the requirements.

Fig. 16. A TeraLogic HDTV Hierarchical chip done by KBPP Methodology

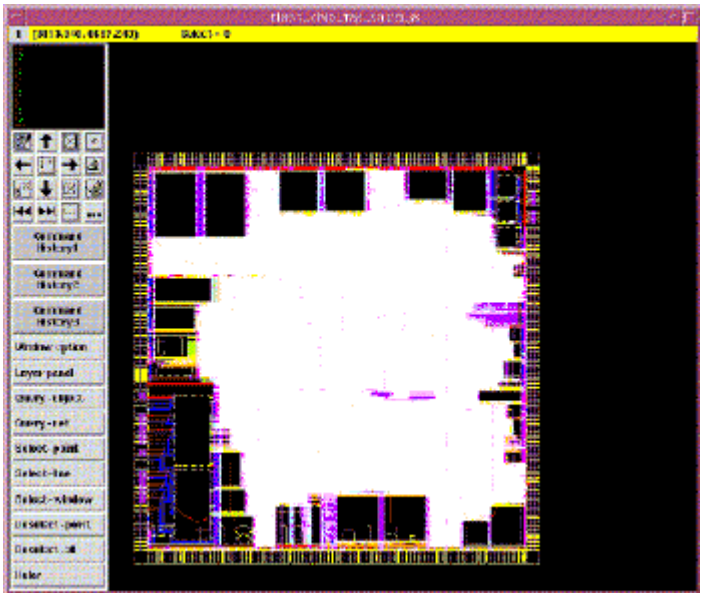
Fig. 17. A TeraLogic HDTV flatten chip done by KBPP Methodology

4.0 Conclusion and Recommendation

This paper proposes the solution for antenna fixing by dynamic dropping the tap (a small and

special kind of protection p-diode and n-diode) as needed during the after the placement and before the routing and a simplified cell placement algorithm. It will be much more convenient for end users if Synopsys's physical compiler and layout tools have these features embedded in their tool floor.

5.0 Acknowledgements

The author would like to thank the following companies and people's help. LSI logic datapath and Sun's Java chip projects gave me the chance to instantiate the implementation and trials of fast placer concept. National Semiconductor Corporation's Audio chip gives me the chance to develop the concept of how to resolve antenna effect. Teralogic Inc's flatten design gives anther trial of the fast placer. My technical writer Stacy Austin, reorganize and review carefully and challenge me with effective questions

2.0 References

[1] D. Beasley, D. R. Bull, and R. R. Martin, "An Overview of Genetic Algorithms: Part 1, Fundamentals," *University Computing*, Vol. 15, pp. 58-69 and pp. 170-181 (1993).

[2] D. Beasley, D. R. Bull, and R. R. Martin, "An Overview of Genetic Algorithms: Part 2, Research Topics," *ibid.*, Vol. 15, pp. 170-181 (1993).

[3] P. H. Chen, S. Malkani, C. M. Peng, and J. Lin, "Fixing Antenna Problem by Dynamic Diode Dropping and Jumper Insertion," *IEEE-ISQED*, pp. 275-282, San Jose, California (2000).

[4] P. H. Chen, "Timing Driven VLSI Physical Design Methodology with Fixing Antenna Effect, Fast Placement, and Timing Convergence", *Ph.D. Dissertation*, International Technological University, Santa Clara, California. Web site http://www.peter-h-chen.com/sys-tmpl/nss-folder/paper/dissertation/phd.pdf.

[5] P. H. Chen, S. Malkani, C. M. Peng, and G. Ying, "Fixing Antenna by Synopsys SLE," *SNUG*, TA4, pp. 1-21, San Jose, California (2000).

[6] P. H. Chen and *et al*, "Chapter 8: Physical Design Exchange Format," *IEEE Standards-P1481 for Integrated Circuit Delay and Power Calculation System*, pp. 265-299, http://www/vhdl.org/vi/dpc (1997).

[7] P. H. Chen and *et al*, "LSI datapath placer implementation specification," LSI Logic Corp, Milpitas, California (1995).

[8] C. K. Cheng and E. S. Kuh, "Module Placement Based on Resistive Network Optimization," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, Vol. CAD-3, No. 3, pp. 218-225, July, 1984.

[9] W. E. Donath, "Placement and Average Interconnection Lengths of Computer Logic," *IEEE Transactions on Circuits and Systems*, Vol. CAS-26, No. 4, pp. 272-277, April 1979.

[10] K. M. Hall, "An r-Dimensional Quadratic Placement Algorithm," *Management Science*, Vol. 17, No. 3, pp. 219-229, November, 1970.

[11] J. M. Kleinhans and *et al*, "GORDIAN: VLSI Placement by Quadratic Programming and Slicing Optimization," *IEEE Transaction on Computer-Aided Design*, Vol. 10, No. 3, pp. 256-365, March 1991.

[12] J-M. Luchies, P. Simon, F. Kuper, and W. Maly, "Relation between Product Yield and Plasma Process Induced Damage," *Proceedings of the P2ID Conference*, pp. 7-10, June, 1998.

[13] R. A. Rutenbar, "Simulated Annealing Algorithms: An Overview." *IEEE Circuits and Devices*, pp. 19-26 (Jan. 1989)

[14] H. C. Shin and C. Hu, "Plasma Etching Antenna Effect on Oxide-Silicon Interface Reliability," *Solid-State Electronics*, Vol. 36, No. 9, pp. 1356-1358 (1993).

[15] H. C. Shin and C. Hu, "Thin Gate Oxide Damage Due to Plasma Processing," *Sci. Technol.*, 11, pp. 463-373, 1994.

[16] K. Shahookar and P. Mazumder, "A Genetic Approach to Standard Cell Placement using Meta-Genetic Parameter Optimization," *IEEE Transaction on Computer-Aided Design*, Vol. 9, No. 5 (May, 1990)

[17] R-S. Tsay, E. S. Kuh, and C-P. Hsu, "PROUD: A Fast Sea-Of-Gates Placement Algorithm," *25th ACM/IEEE Design Automation Conference*, pp. 318-426 (1988).