Global Routing for Three Dimensional Packaging

Jacob R. Minz and Sung Kyu Lim
School of Electrical and Computer Engineering
Georgia Institute of Technology, Atlanta, GA 30332-0250
{jrminz,limsk}@ece.gatech.edu

ABSTRACT

Three dimensional packaging is becoming a popular concept because of the numerous advantages it has to offer over the existing conventional technologies. System on Packages (SOP) is an example of three dimensional packaging. The contribution of this work is threefold: (i) formulation of the new 3-dimensional global routing problem, (ii) a new routing flow that considers the various design constraints unique to SOP, and (iii) a global router for the technology. Our related experimental results demonstrate the effectiveness of our algorithm.

1. INTRODUCTION

The true potential of SOP technology lies in its capability to integrate both active components such as digital IC, analog ICs, memory modules, MEMS, and opto-electronic modules, and passive components such as capacitors, resistors, and inductors all into a single high speed/density multi-layer packaging substrate. Since both the active and passive components are integrated into the multi-layer substrate, SOP offers a highly advanced three-dimensional mixed-signal system integration environment. Three-dimensional SOP packaging offers significant performance benefits over the traditional two-dimensional packaging such as PCB and MCM due to the electrical and mechanical properties arising from the new geometrical arrangement as illustrated in Figure 1. Thus, innovative ideas in the development of CAD tools for multi-layer SOP technology is crucial to fully exploit the potential of this new emerging technology.

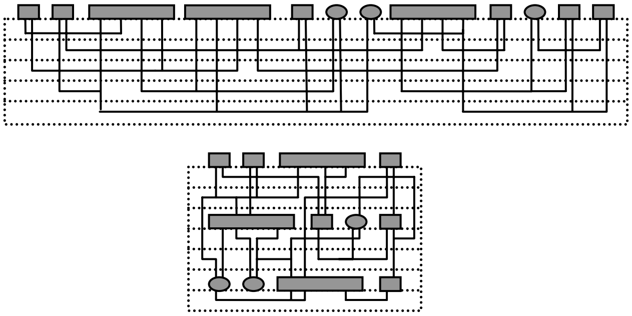

Figure 1. Single active layer (PCB,MCM) vs multiple active layer (SOP) packaging.

The physical layout resource of SOP is multi-layer in nature—the top layer is mainly used to accommodate active components, the middle layers are mainly for passive components, and the I/O pins are located at the bottom of the SOP package. Routing layers are inserted in between these floorplan layers, and the floorplan layers can be used for local routing as well. Therefore, all layers are used for both floorplan and routing and pins are now located at all layers rather than the top-most layer only as in PCB or MCM. Therefore, the existing routing tools for PCB or MCM [4,5] can not be used directly for SOP routing.

In this paper, we propose a new interconnect-centric global routing paradigm that handles arbitrary routing topologies and produces good results. The contribution of this work is threefold: (i) modeling of the SOP routing resource, (ii) formulation of the new SOP global routing problem, and (iii) development of a fast and novel algorithm that considers the various design constraints unique to SOP. We review various approaches for the PCB, IC and MCM algorithms and investigate their applicability to the SOP model. Our related experimental results demonstrate the effectiveness of our approach.

2. SOP GLOBAL ROUTING PROBLEM

2.1 Routing Resource Model

The layer structure in SOP is different from PCB or MCM—it has multiple floorplan layers and routing layers. It has one *I/O pin layer* through which various components can be connected to the external pins. The *floorplan layers* contain the blocks, which from the point of view of physical design is just a geometrical object with pins. In some cases where these blocks are a collection of cells, the pins may not be assigned and pin assignment needs to be done to determine their exact location. The interval between two floorplan layers is called the *routing interval*. The routing interval contains a stack of *signal routing layers* sandwiched between *pin distribution layers*. These layers are actually X-Y routing layer pairs, so that the rectilinear partial net topologies may be assigned to it. We also allow routing to be done in the pin distribution layers.

We model the floorplan layer in the SOP as a floor connection graph [2]. The routing layer is modeled as a uniform grid graph. These two kinds of graphs are connected through via edges. The advantage of our graph-based routing resource model is that we can consider layer/pin assignment and global routing simultaneously. We model the blocks in the floorplan as *Block Nodes* (BN). The nets can cross over to the adjacent routing layers only through the regions in the channel. The channel itself is represented by *Channel Nodes* (CN). The actual blocks form blockages for the nets, which cannot be routed through them. The nets can switch from floorplan layer to the routing layer only through designated regions which are represented as *Layer-switch Nodes* (LN) in the resource graph. The LN in this case is simply four corners of the blocks. They denote regions rather than points through which nets will traverse to adjacent routing intervals. The routing layers are represented by a grid graph, each node specifying a region in the layer and edges representing the adjacency between regions. These nodes are called *Routing Nodes* (RN). The concepts are illustrated in Figure 2, which shows the various views and types of nodes used in SOP.

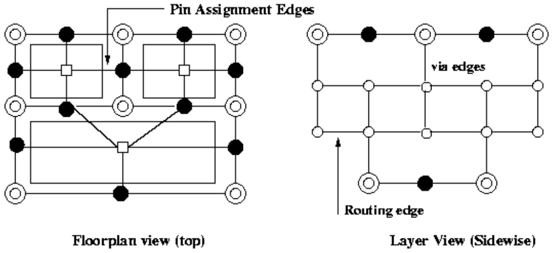

Figure 2. The floorplan and the layer view of the resource graph with node and edge types. Double circle, square, black dot, and white dot respectively denote the layer switch, block, channel, and routing nodes.

The edges between channel nodes and block nodes are called *Pin Assignment Edges* (PE). This makes it possible to perform pin assignment during global routing. The pin assignment capacity is the maximum number of pins which can be assigned towards a particular channel. The edges between layer switch node and routing node is defined as *Via Edges* (VE). The capacity of this edge is the maximum number of nets which can cross between two regions in the two layers. The via edges also exist between two adjacent routing layers (actually layer pairs). The edges between routing nodes are *Routing Edges* (RE). The routing edge capacity is the number of nets which can pass through the routing regions.

2.2 Global Routing Problem Formulation

We define the SOP global routing problem formally as follows: Given a set of floorplans $F=\{f_1,f_2,\ldots,f_k\}$, netlist $N=\{n_1,n_2,\ldots,n_n\}$, and the routing resource graph, generate the routing topology $T(n)$ for each net n, assign n to a set of routing layers and assign all pins of n to legal locations. All conflicting nets are assigned to different routing layers while satisfying various capacity constraints. The objective is to minimize the total number of routing layers used, wirelength, and crosstalk.

In the SOP model the nets are classified into two categories. The nets which have all their terminals in the same floorplan are called i-nets, while the ones having terminal in different floorplans will be referred to as x-nets. The i-nets can be routed in the single routing interval or indeed within the floorplan layer itself. However, for high performance designs routing such nets in the routing interval immediately above or below the floorplan layer maybe desirable and even required. On the other hand, the x-nets may span more than one routing intervals. The only case where one routing interval may suffice is when the terminals of the net are located in either of the floorplans immediately above or below the routing interval. The span of a net $[l, h]$ is determined by the lowest floorplan f_l and the highest floorplan f_h containing pins of the net. If l and h are equal for a particular net, the net is i-net else the net is x-net.

3. SOP GLOBAL ROUTING ALGORITHMS

3.1. Overview of the Algorithm

Our SOP router is multi-phased, where we divide the routing process into (1) coarse pin distribution, (2) net distribution, (3) detailed pin distribution, (4) topology generation, (5) 2D layer assignment, (6) channel assignment, and (7) pin assignment

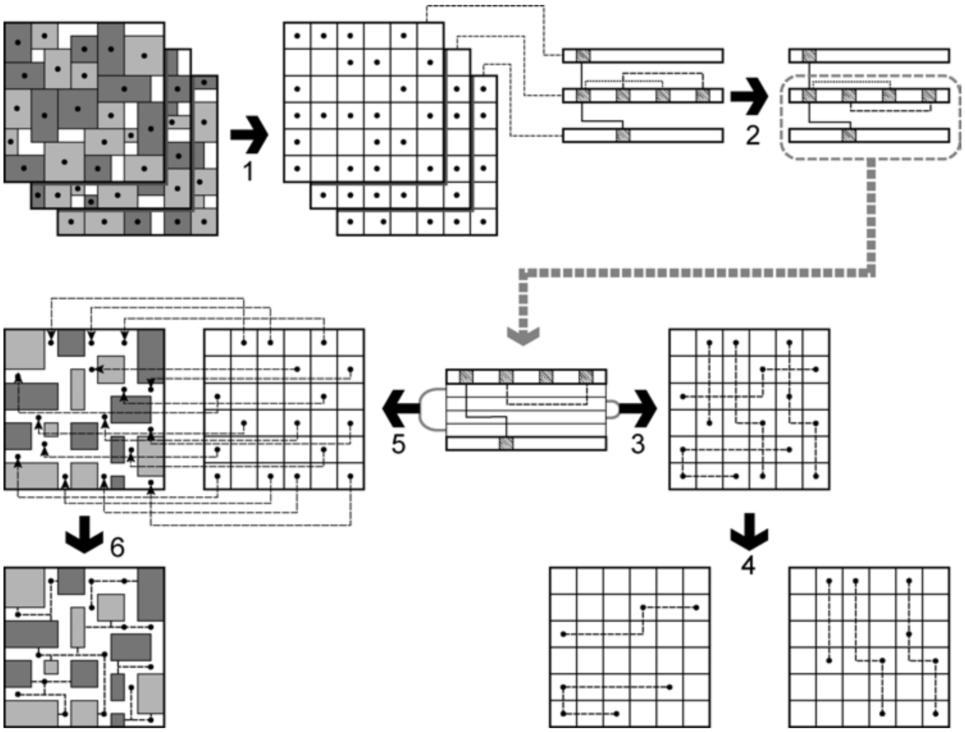

Figure 3. Overview of the global routing process. 1=pin distribution, 2=net distribution, 3=topology generation, 4=layer assignment, 5=channel assignment, 6=pin assignment.

step. An illustration is shown in Figure 3. By following these steps we seek to have enough and acceptably accurate information for each routing interval to carry out “local” global routing efficiently. We introduce the terms entry/exit pins to better explain our approach. Since the connections of some nets (x-nets) span multiple placement layers, we need to determine the location of entry to and exit from the routing interval. In the 2-D refinement of the problem we treat this location as pins. The routing of i-nets deserves special attention. In routing intervals, except the first and last ones, we have the choice of placing those i-nets in a routing interval either on top or bottom of the floorplan. The objective is to minimize crosstalk and congestion in the routing interval. This step is called *net distribution*.

The pin information of the nets is required for efficient net distribution, but net distribution decides the number of pins (and their locations) at each routing interval. We solve this by using the results of *Coarse Pin Distribution* for net distribution. Pins in all routing interval are projected to a single 2-D area and partitioning evenly distributes them over it. The pins in different routing intervals may not be evenly distributed locally. After net distribution we do *Detailed Pin Distribution* on each routing interval to minimize the estimated wirelength and legalize pin locations. After this step we have all the information needed for global routing in the routing interval. The topology of each net is generated during our *Topology Generation*, and *2-D Layer Assignment* assigns different layer to conflicting nets. The *Channel Assignment* problem is to assign each pin in the pin distribution layers to a channel in the floorplan layers such that the routing layers and interconnect costs are minimized. The objective is to facilitate an efficient pin distribution on pin distribution layer with only minimal additional costs. The purpose of *Pin Assignment* is to assign a location to the pin on the block boundary on the floorplan layer while minimizing the connections between the pin and its “peer” on the channel, which was found out in the previous step. The peer is location in the floorplan which connects the net to rest of its interconnect in the routing layers. Figure 3 illustrates the main ideas of the algorithm.

3.2 SOP Global Routing Algorithm

The nets may be provided as connections between blocks in which case we may have to do *pin assignment* on the blocks. *Pin distribution* can be done irrespective of whether or not pins have been assigned on the block because all that is relevant to the problem at hand is the entry/exit points to the routing interval. This is done as soon as the pins are generated. For the purpose of the algorithm we define two types of nets, *current* and *propagated* nets. We visit the floorplans from the lowest to the highest level, in other words we consider the routing intervals sequentially from lowest to highest. The current nets are those which will be considered for layer assignment in the current routing interval. The current routing interval is the one currently processed by the algorithm. The propagated nets are nets “passed on” from this interval to be considered in the next routing interval. For example x-nets will be propagated from its lowest level to the highest and will also be the current net for all the routing intervals in between. This is because we consider only a part (segment) of the x-net for routing in a particular routing interval (x-nets span multiple routing intervals). In the case of i-nets, the net is either current or propagated. This is found out by *net distribution*. The propagated nets form a subset of the nets to be routed in the next routing interval to be processed, since some i-nets may be included in the next routing interval. 2-D global routing simultaneously with layer minimization is done for each routing intervals. Channel assignment connects pins from the routing interval to the placement layers on allowed spots. Pin assignment connects the blocks to the corresponding vias in the channel with minimum interconnection costs. The last two steps are done once for each placement layers.

3.3 Summary of Our Recent Work

We recently have implemented the Coarse Pin Distribution and 2-D Layer Assignment [6] and Net and Pin Distribution [7] for 3D packaging layout. For topology generation, we have used an existing RSA/G heuristic [3] to generate the net topologies at each routing interval, since it is fast and simple. The minimum arborescence is a good representative for the topology of a net in a high performance design.

Coarse Pin Distribution: In this step, we generate coarse locations for all pins of the nets in the routing interval. For the purpose of pin distribution we “flatten” the 3-D SOP structure to 2-D and superimpose a AxB grid on it, were A and B are determined by the size of the circuit. We use our partitioning algorithm [8] to evenly distribute pins to all the partitions formed by this grid while keeping the wirelength minimum. Evenly distributing the pins among all partitions ensures efficient use of the routing resource provided by the single layer. The “coarse” location is the centre of the partition. After the partitioning the pins may not be uniformly distributed in the local routing interval. This partitioning algorithm is smart enough not to move the pins far from their “initial” locations. The algorithm does iterative improvement until good results are obtained.

2-D Layer Assignment: We construct a Layer Constraint Graph (LCG) from the given global routing topology, where each node represents a net and two nodes in the LCG have an edge between them if corresponding net segments of same orientation (horizontal or vertical) share at least one tile in the routing grid. We use a fast node coloring heuristic algorithm to assign a color to the node such that no two nodes sharing an edge are assigned the same color. The algorithm is greedy in assigning colors but performs well and is fast. Close to lower bound results are achieved because the heuristic tries to ensure that nodes with different colors have in fact an edge between them. The complexity of the algorithm is $O(n\log n)$, where n is the number of nets in the routing interval. The complexity is independent of the size of the grid used to compute the tree topologies. The capacity of the tiles determines the number of layers used. We use a simple formula to calculate this number (number of colors/capacity).

Net Distribution: Net assignment for some nets is straight forward. When the floorplans are visited bottom to top, all nets having their pins in the lowest floorplan are assigned to the routing interval above it. The nets having pins the top-most floorplan are assigned the routing interval right below it. If the net is an x-net it is propagated through every layer until its topmost floorplan is reached. The net distribution of the i-nets is interesting. The objective of this step is to reduce crosstalk. We use the amount of overlap of bounding boxes of the nets as a measure of crosstalk. The net distribution problem is modeled as a graph with each i-net in the routing interval as node and the crosstalk interaction as edges. The weight of the edges denotes the amount of crosstalk between the nets. The crosstalk is calculated by the overlap of the bounding boxes of the net. The coarse pin distribution is used as the approximate location of the pins. It is assumed that nets in different interval are crosstalk shielded, which means no crosstalk exist between nets in different interval. The problem can then be seen as a restricted graph partitioning problem where some of the nodes can only go to one of two predetermined partitions. In order to achieve better results iterative technique in [8] are used. The complexity of the algorithm is O(V+E) where V is number of nodes and E is the number of edges in the graph model. The results obtained were compared with random net distribution and the case when no i-nets are propagated to other routing intervals.

Detailed Pin Distribution: The purpose of this step is to legalize the location of the pins while respecting the coarse pin assignment and optimizing wirelength. The results of the coarse pin assignment are used for force-directed placement of the pins in the pin distribution layers. Since we did not consider the layer in which the pin was located in the coarse pin redistribution, it may be possible that the pins exceed the capacity of the partitions local to the routing interval. However our algorithm handles this by moving the pins from such location to the closest available position. The pins are placed in locations near the centre of the net. The pins furthest from its center of the net in coarse assignment, gets placed in the best location (location nearest to the center) in the local partition. The algorithm uses the “approximate” position of the pins as found by coarse pin distribution and the net distribution results to determine the initial location and routing interval of the pin. The position of the pins is stored as the grid location of the coarse pin distribution. The center of each net is calculated from this position of the pins. The displacement vector is calculated by taking the difference of the position of the center of net and the pin. A pair of numbers (a,b) such that $0<a<1$, $0<b<1$ is added to the position of the pins. The numbers reflect the scaled magnitude of the displacement vector. The variables a and b are less than 1 so that we can still keep track of the partitions of the pins. The pins in each routing interval are sorted according to their new positions. The pins are then sequentially assigned to grids previously determined.

3.4. Contribution of this Paper

In this section we explain in detail the algorithms used to handle channel assignment and pin assignment in the floorplan layer. The objectives we tried to minimize are the number of additional layers for pin redistribution during channel assignment and the total wirelength during pin assignment.

3.4.1 Channel Assignment

The pins in the routing interval have to be connected to their corresponding blocks in the floorplan layer or if no such block exist, to its counterpart pin to the other routing intervals (for x-nets). The pins are connected to the floorplan layer using vias which can only be accommodated in the routing channels. The pins therefore have to be assigned a channel in the floorplan layer. The channel assignment of pins will affect the additional number of layers and total wirelength. Since one desirable objective is to reduce the number of bends (which would necessitate the use of secondary vias), we assume a straight or L-shaped routing of nets to their assigned channel. This reasonable assumption simplifies the evaluation of the wirelengths. We observed that congestion of pin connections and wire crossings on a particular channel would increase the layer count. Our cost model for the problem captures these issues and our algorithm minimizing the cost function, assigns every pin to a channel. For this problem we are given the location of the pins and the location, length, orientation (horizontal or vertical) and via capacities of the channel. We seek to minimize the additional number of layers and wirelength while assigning every pin to a channel.

```
Algorithm:  CHLDIST
Input: pins, channels
Output: channel assignment & number of layers
Determine via capacities for each channel.
For all pins
    Find a channel with minimum assignment cost
          and not violating the constraints.
    Calculate channel crossings.
    Update via and routing demands of all channels.
    Assign this channel to the pin.
Calculate and report number of layers.
```

Figure 4. SOP channel assignment algorithm

We derive the number of layers as follows. We assume that the channels will only accommodate rectilinear routes *perpendicular* to their orientation. For example, only vertical routes can terminate at or cross horizontal channels. This is a reasonable assumption because routes can still go parallely over that channel; however they terminate at some other channels. We note that both terminating and crossing routes on the channel affects the routing demands on the channel. The routing demands were classified into left (top) or right (bottom) demands for vertical (horizontal) channels. Let the maximum of the two routing demands divided by the channel capacity be *den*. Then the number of layers for the channel assignment is given by the maximum density among all channels.

The algorithm (Figure 4) for channel assignment assigns channels to the pins based on the costs of channel assignment. The cost is the sum of L-distance between pin and channel, the channel density and the bending penalty, multiplied by constants to reflect the relative importance. However, the constants are finalized by trying different values for a particular benchmark and finalizing it for all experiments. The L-distance and bending penalty between channel and pin is constant part of the cost while the channel density needed to be updated with an assignment.

3.4.2. Pin Assignment

The final step in our proposed methodology connects the nets to its "original" terminals. The pin assignment is done entirely in the floorplan layer. Since the connections of the nets are specified by a set of blocks, the location of the terminals on the block boundary is to be determined. The routing channel is used to finish the last connections from the channel pins (determined during channel assignment) to the block. The channel pins are actually the entry/exit points to the routing interval. An interesting aspect of this problem is that *complete* connections of the blocks and channel pins of a net is *not* necessary since the channel pins of a net are connected in the routing interval. Hence it suffices if the block is connected to at least one channel pin. This observation reduces the problem to a 2-terminal net pin assignment. We model the floorplan with a Floorplan Connection Graph. The pin is now either a block node or channel node. We use modified Djikstra's algorithm to find the most feasible coarse location for the terminals on the block boundary. The pseudocode is given in Figure 5.

```
Algorithm:  2PINASSIGN
Input: 2-D floorplan, netlist
Output: routes and course pin assignment
Generate FCG from the floorplan.
Generate 2-pin subnets for all nets in this floorplan.
Initialize the dynamic cost of the edges.
For all 2-pin subnets
    Find shortest path between source & sink   node.
        Update the cost of edges on the path.
Rip-up and reroute when dealing with capacities.
```

Figure 5. SOP pin assignment algorithm

The key to efficiently do pin-assignment for 3-D packaging is to have a good 2-pin net generation. The pins which had been projected to routing intervals during pin generation for routing interval now needs to be connected to its originating blocks. However, the pins can also connect to blocks closer to them, which form the part of the same net, if the costs are improved. The edge weights of the FCG derived from the floorplan are initialized. In the proposed solution we try to minimize the demands on routing as well as the pin-assignment edges while determining the path between the source and destination nodes. We force selection of different routes by making the costs of the edges in the path *high*. This ensures fair usage of resources modeled by the edges.

4. EXPERIMENTAL RESULTS

We implemented our algorithm GROUTE in C++/STL and ran experiments on a Dell Dimension 8800 Linux box. Our test cases are generated using our multi-layer SOP floorplanner on GSRC benchmark circuits [9]. The number of layer is fixed to four. Our layer usage results are based on the tile density w=10. The RSA/G-based global routing trees are generated based on 10×10 unless otherwise specified. We ran our algorithms for channel assignment and pin assignment on GSRC benchmarks and measured its solution quality as the values obtained for the various objectives. For our initial experiments, we have not taken the I/O layer into consideration. All the benchmarks completed in less than a few minutes. So we do not explicitly report the runtimes.

4.1. Channel Assignment Results

We present the results of our Channel Assignment algorithm in Table 1. In order to compare the quality of the solutions achieved by our channel assignment algorithm, we computed the best possible wirelength for a channel assignment where via capacity violations were allowed. We tabulate the results of this scheme under the best wirelength. We compare the results of our algorithm with the best wirelength results for the number of layer pairs, wirelength, bends and number of pins violating channel via capacities. Our inferences based on the experimental results are:

1. In channel assignment, we are close to the number of layers predicted by the best case. The increase in layers is due to increased routing density on the channels. The ratios of actual wirelength with best wirelengths increase with the size of benchmarks.
2. The violations in the best case and the number of bends reported by our algorithm are very close, suggesting that violations were fixed by bending the interconnections.

Table 1. The result of multi-objective minimizing channel assignment. Number of layer pairs(ly), wirelength(wl), bends (bnd) and violations(vl) for the best and actual cases are reported.

Ckt	*Best Wirelength*				*CHASSIGN*			
	ly	*wl*	bnd	*vl*	ly	*wl*	bnd	*vl*
n10	5	6963	0	55	5	14174	34	0
n30	6	11288	0	144	6	24726	122	0
n50	6	14826	3	282	6	32059	189	0
n100	6	16430	3	413	7	36331	382	0
n200	6	24078	1	845	8	61485	917	0
n300	6	28469	3	1000	8	65189	1029	0

4.2 Pin Distribution Results

In Table 2 we report the wirelength achieved during pin assignment. The result of the channel assignment is used as input to the pin assignment. For generating the best wirelength, we used the corresponding best wirelength channel capacities violating channel assignment. For the best wirelength, we allowed pin assignment algorithm to select routes without considering the pin assignment and routing demands. Our algorithm tries to minimize wirelength while avoiding congestion of routing channels and pin assignment resources. The parameters of the algorithm decide the trade-off between wirelength, pin assignment demands and routing demands. From our experiments we observe the following:

1. In pin assignment, we are able to reduce pin assignment and routing demands drastically by increasing only 25% wirelength, from the best case.
2. The wirelength scales rapidly with benchmark sizes and the wirelength for pin assignment is huge compared to channel assignment due to limited routing resources in the floorplan layer.

Table 2. The pin assignment results. Wirelength(wl), pin assignment demands(pd) and routing demand(rd) are reported.

Ckt	*Best Wirelength*			*2PINASSIGN*		
	wl	pd	rd	*wl*	pd	rd
n10	10327	11	15	12173	9	11
n30	42108	16	47	54730	12	18
n50	86052	31	70	120743	15	59
n100	155089	20	83	194724	12	42
n200	343883	23	139	424620	11	76
n300	561244	27	162	692383	11	79

5. CONCLUSIONS

In this paper we discuss issues related to global routing for 3-D packaging. We discuss the methodology used in our global router for SOP. We have presented the results of channel and pin assignment which constitutes the final steps in our proposed routing flow for the SOP. To the best of our knowledge, ours is the first implemented global router for SOP addressing objectives such as crosstalk, wirelength, and total number of layers.

6. REFERENCES

[1] Rao Tummala and Vijay Madisetti, "System on Chip or System on Package?", *IEEE Design & Test of Computers*, pp 48-56, 1999.

[2] J. Cong, "Pin assignment with global routing for general cell designs", *IEEE Trans on Computer-Aided Design of Integrated Circuits and Systems*, pp1401-1412, 1991

[3] J. Cong, A.B. Kahng, K Leung, "Efficient Algorithms for the Minimum Shortest Path Steiner Arborescence Problem with Applications to VLSI Physical Design", *IEEE Trans. on CAD*, vol. 17, no.1, January 1998

[4] K.-Y. Khoo and J. Cong, "A fast multilayer general area router for MCM designs," *IEEE Trans. Circuits and Syst. II*, Nov. 1992, vol.39, pp. 841-51

[5] K.-Y. Khoo and J. Cong, "An efficient multilayer MCM router based on four-via routing," *IEEE Trans. Computer-Aided Design*, Oct. 1995, vol.14, pp. 1277-90

[6] Omitted.

[7] Omitted.

[8] Omitted

[9] http://www.cse.ucsc.edu/research/surf/GSRC/
GSRCbench.html