Full-Chip Scalable Routing Framework Considering Congestion and Performance*

Yao Hailong, Cai Yici, Hong Xianlong†, and Zhou Qiang

(*Department of Computer Science & Technology, Tsinghua University, Beijing* 100084, *China*)

Abstract: This paper presents a novel full-chip scalable routing framework that simultaneously considers the routing congestion and the circuit performance. In order to bridge the gap, the presented framework calls the detailed router immediately after a global route is extracted. With the interleaving mode of global routing immediately followed by detailed routing, accurate routing resource and congestion information can be obtained, which provides valuable guidance for the following global routing process. The framework features the fast pattern and framed shortest path global router, a maze-based congestion-driven detailed router, and better interaction between the two routers. In the framework, timing critical nets can be assigned higher priority for performance concern, and different net ordering techniques can be adopted for different routing objectives. The framework is tested on a set of commonly used benchmark circuits and compared with a previous multilevel routing framework. Experimental results show that the presented framework obtains significantly better routing solutions than the previous one considering circuit performance, routing completion rate, and runtime.

Key words: congestion; multilevel routing; performance; scalable routing

EEACC: 2570 **CCACC:** 7410D

CLC number: TN47 **Document code:** A **Article ID:** 0253-4177(2006)07-1201-08

1 Introduction

As VLSI technology reaches very deep submicron feature size and gigahertz clock frequencies, the timing issue is becoming more critical. The interconnect has become the dominating factor in determining the performance of the overall chip. Thus, the routing has become a more challenging problem. The objective of routing is not only to accomplish a high completion rate within an endurable runtime but also to satisfy other constraints, such as timing and crosstalk.

Traditionally, due to the high complexity of the routing problem, it has been solved by the divide-and-conquer approach in two sequential stages: global routing and detailed routing. In literatures, many algorithms in global and detailed routing have been presented for the above-mentioned two-stage flat routing framework. Those algorithms fall into two categories: sequential and concurrent methods. A sequential method routes nets one by one in a predefined order based on maze-searching[1,2] or line-probe[3] techniques. A concurrent method routes nets simultaneously with negotiation-based iterative approaches[4,5] or flow-based methods[6,7].

As VLSI technology advances, the traditional two-stage flat routing framework confronts the scaling problem for large industrial designs. As a result, hierarchical approaches are adopted to handle large routing problems[8~10]. The drawbacks of hierarchical approaches are the constraint of higher level routing solutions on lower level ones and the lack of detailed routing information at higher levels[11].

In Ref.[12], the first multilevel routing framework was proposed and proven to be more effective for large routing problems than the traditional two-stage flat framework and the hierarchical approach. In Ref. [11], some improvements were made to make the multilevel framework more complete and effective. However, this framework is based on gridless detailed routing. In Ref.[13], another multilevel routing (MR) framework based on grid routing was proposed. Unlike Ref. [12], the

* Project supported by the National High Technology Research & Development Program of China (No. 2005AA1Z1230)

† Corresponding author. Email: hxl-dcs@mail.tsinghua.edu.cn

Received 3 November 2005, revised manuscript received 14 February 2006

grid-based multilevel routing ,framework integrates global routing ,detailed routing ,and resource estimation during the coarsening stage. MR proves successful in solving the routing problem with a high routing completion rate and a short runtime. Then in Refs. [14～16] ,some improvements were made to adapt the framework for routing considering performance ,antenna effects ,and X-based architecture. In Ref. [17] ,the framework was adopted for gridless routing considering optical proximity correction. In spite of the great success of MR , there are some insufficiencies in the basic framework considering the timing and performance issue ,which will be discussed in the next section.

Inspired by the previous works on multilevel routing ,we present in this paper a novel grid-based full-chip scalable routing framework (SRF) that takes congestion and performance into consideration. Compared with MR ,our SRF has the following distinguishing features:

(1) Two routing stages ,trying and then rerouting ,are used in SRF without coarsening and uncoarsening steps. First , we decompose all the nets into two-pin subnets by the minimum spanning tree (MST) algorithm ,with each subnet corresponding to one edge of the MST. Then we push the subnets into a priority queue using the criticality-driven least flexibility prior net ordering technique. During the trying stage ,we route the subnets one by one in the queue order with fast global routers followed by a congestion-driven detailed router. The fast global routers include a pattern router and a framed shortest path router , which finds a global route for each subnet with minimum wire length and congestion in the frame bounded by the pins. Each time a subnet is routed ,the corresponding routing resources are updated immediately. During the rerouting stage ,the failed subnets are routed by the framed shortest path router and then the detailed router. The frame of the shortest path router is increased little by little until the subnet is routed or the entire routing area is covered. The scheme in which the global router is immediately followed by the detailed router results in better interaction between global routing and detailed routing. Thus an accurate resource estimate can be obtained ,which greatly benefits future global routing decisions.

(2) The criticality-driven least flexibility prior net ordering technique is introduced to route the nets considering routing completion as well as timing and performance.

(3) A grid-base detailed router is adopted with some optimization techniques such as congestion-driven maze backtrace and point-to-path maze-searching.

With the net ordering technique , SRF gives special concern for timing and reduces the delay of the timing-critical nets. With much better timing results and little loss in other objectives , SRF is promising.

2 Detailed view of MR

Now we have a detailed view of the MR presented in Ref. [13]. In MR ,there are some drawbacks in performance. MR routes the two-pin subnets in a sequential order ,which is basically small net first. Since timing is critical in very deep submicron technology ,long and critical nets should be special concerned in routing to meet the timing constraint. Thus the small net first ordering scheme is not reasonable or suitable for performance-driven routing in current VLSI designs. The impact of the net ordering scheme can be concealed when the routing resources are abundant. However ,when the resources are limited ,the defect will emerge ,resulting in longer delay or even failure in routing for the long and critical nets.

We downloaded the source code of MR from http://cc. ee. ntu. edu. tw/～ywchang/research. html and tested it on commonly used benchmark circuits. Table 1 shows the routing results of the critical nets. The "Circuit" column gives the names of the circuits , whose detailed information is shown in Table 3. Here we only use two routing layers to make the problem more clear. We select a random number of the longest nets ,whose pins occupy the largest areas ,and mark them as critical. Since information about the critical nets or the longest nets is not available before the detailed routing is finished ,we have to adopt such heuristics to choose the possible critical nets. However ,it is easy to understand that in this way ,we can always select the long nets. The "Crit. Ratio" cloumn gives the ratio of the wire length of critical nets to the total wire length after routing in the SRF system. We use the

SRF system to calculate the data because SRF can route 100 % of all the test examples, whereas MR cannot. The " # Net " column gives the total number of nets, and the " # FN " cloumn gives the number of failed nets in MR routing. The " # FCN " column denotes the number of failed critical nets, and the " Ratio " column shows the ratio of the number of failed critical nets to the number of total failed nets. From Table 1, we can see that except for Primary 1, Primary 2, and Struct, whose routing resources are still abundant, critical nets take up to 42.9 % of the total failed nets for other test examples. Even for the 100 % routed circuits, the wire lengths of the critical nets must be longer than they ought to be. This is because the long and critical nets are routed after the small nets, and they probably have to detour because of the pre-routes. However, the critical nets with enlarged wire length will probably violate the timing constraint, which is forbidden in current VLSI design.

Table 1　Routing results of MR for critical nets

Circuit	Crit. Ratio / %	# Net	# FN	# FCN	Ratio / %
Mcc1	26.4	1694	8	3	37.5
Primary 1	10.7	2037	0	0	—
Primary 2	17.2	8197	0	0	—
S13207	30.5	6995	15	3	20.0
S15850	31.7	8321	21	9	42.9
S38417	42.4	21035	28	5	17.9
S38584	40.6	28177	50	4	8.0
S9234	23.6	2774	8	1	12.5
Struct	38.7	3551	0	0	—
S5378	17.2	3124	21	7	33.3

Now we explain why net ordering in MR is basically small net first. Table 2 shows the net routing condition of MR on the original circuits with unreduced routing layers. The " # Subnet " column denotes the number of two-pin subnets after MST-based net decomposition. The " Coarsening " column shows the number of routed subnets in the coarsening stage, and the " Uncoarsening " column gives the number of failed subnets during coarsening, which need to be refined during uncoarsening. From Table 2, we can see that most nets are routed in the coarsening stage, and only a few failed nets remain for the uncoarsening stage.

Table 2　Nets routed during coarsening of MR

Circuit	# Subnet	Coarsening	Uncoarsening
Mcc1	1694	1693	1
Primary 1	2037	2037	0
Primary 2	8197	8197	0
S13207	6995	6995	0
S15850	8321	8320	1
S38417	21035	21034	1
S38584	28177	28176	1
S9234	2774	2774	0
Struct	3551	3551	0
S5378	3124	3120	4

However, during the coarsening stage, detailed routing is done on each level in a sequential order defined by the coarsening process. Considering the characteristics of the coarsening process, the defined net ordering is basically small net first. However, there are exceptions. In Fig. 1, we can see that although net n_1 is a small net, because of its special location in the routing area, it is routed at the end of the coarsening stage. On the other hand, since net n_2 is on the routing tiles that will first be merged during the coarsening stage, it will be routed prior to net n_1. Thus, nets are ordered according to their relative positions on the routing tiles, not just small net first. For the above-mentioned reasons, we draw the conclusion that the primitive net ordering scheme in MR is not suitable for current routing, which must take into consideration timing performance.

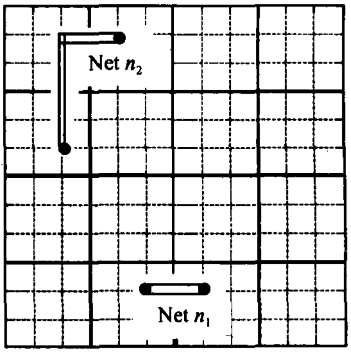


Fig. 1　Net ordering during coarsening stage of MR

3　Net ordering in SRF

In light of the observation in Section 2, we adopt a more reasonable and effective net ordering technique that gives special concern to the critical nets to improve the timing and the performance of the circuits. In the SRF system, nets are first decomposed into two-pin subnets using the minimum spanning tree (MST) algorithm. Then the subnets are routed one by one. In order to achieve a high completion rate and satisfactory timing results, we sort all the subnets into an appropriate order. The sorting criterion considers the following two factors:

(1) Critical nets first

Since long nets often run into timing problems, we give them higher priority for routing. In the implementation, we select a random number of the longest nets as critical nets and route their corresponding subnets first to reduce the delay. We also regard other nets which need special concern as critical nets. Thus the optimal routing solution for the nets with special optimization requirements can be obtained.

(2) Least flexibility first

This criterion deals with the difficulty of routing a subnet. The freedom of a subnet is defined as the weighted sum of the size of the horizontal minimum cut set and the vertical minimum cut set in the undirected grid graph within the rectangular area formed by the two pins of the subnet. Figure 2 illustrates the minimum cut sets of subnet *n*. In Fig. 2, the size of the horizontal minimum cut set (*hc*) is 5, and the size of the vertical one (*vc*) is 9.

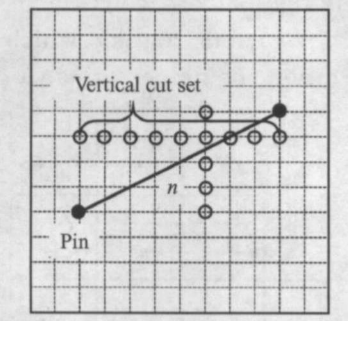


Fig. 2 Minimum cut set of a subnet

The smaller the cut set size is, the more difficult it is for the subnet to be routed with optimal wire length. We formulate this notion of the freedom of a subnet, which can be calculated as follows:

$$\text{freedom} = \alpha \times \min(hc, vc) + \max(hc, vc) \quad (1)$$

where α is a user-defined parameter which makes the first term min (*hc*, *vc*) dominate the second one. Then we assign each subnet a priority which is computed as follows:

$$\text{priority} = \beta \times \text{criticality} - \text{freedom} \quad (2)$$

where β is defined by the user and makes the first term dominate the second one. The criticality is 1 or 0, according to whether the subnet is critical. We can see that the value of priority is positive for critical nets and negative for others. With the priority calculated for each subnet, we can build a priority queue and route the subnets according to the queue order.

4 Global router

In the SRF system, there are two stages for routing a subnet. The first one tries to route a subnet with minimum wire length, so we call it the trying stage. The second one is the rerouting stage, during which failed subnets from the trying stage are rerouted, possibly with longer wire length.

During the trying stage, we route the subnets in the predefined order described in Section 3. For each subnet, we first try to route it using the global router if it is not a local net within one routing tile. Then the global router calls the detailed router to finish the routing. The global router runs in a certain rerouting pattern. When one global routing result is not feasible, another result will be extracted from the global router and fed to the detailed router for rerouting. The rerouting process continues for several iterations until detailed routing succeeds or the predefined number of iterations is reached.

The global routers in the trying stage include the line-pattern router, U-pattern router, L-pattern router, Z-pattern router and the framed shortest path (SP) router. The pattern routers are based on those in Ref. [18]. Line- and U-pattern routers are used for line-shape subnets, whereas L- and Z-pattern routers are for L-shape subnets. Each router calls the detailed router to finish routing. If one router fails, the following router will be called for a further try. Figure 3 illustrates the four pattern-based routes. When pattern routers fail, we use the framed SP router, which is based on the shortest path algorithm[19]. Both the Z-pattern router and the framed SP router need a cost function for guidance in path searching. Let $G = (V, E)$ be the routing graph of the frame bounded by pins of the subnet and $R(n) = \{e | e \in E$ and e is selected for the route of $n\}$ be a global route for subnet *n*. Then the cost for the global route $R(n)$ can be calculated as

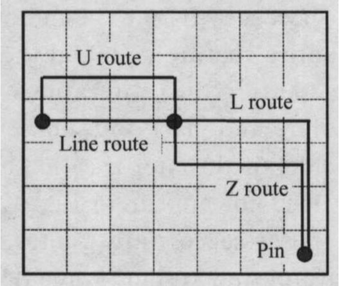


Fig. 3　Pattern-based routes

follows:

$$c(R(n)) = \sum_{e \in R(n)} f(e) \tag{3}$$

Here $f(e)$ is the cost function for using the edge e. The cost function is computed as follows:

$$f(e) = a/2^{d_e - c_e} + b/2^{wd_e - wc_e} + c/2^{td_e - tc_e} + dl_e + fo_e \tag{4}$$

Here c_e and d_e are the capacity and density of edge e, respectively, wc_e and wd_e are the wiring capacity and wiring density of the routing tile associated with edge e, respectively, which measures the routing congestion in the tile, tc_e and td_e are the through capacity and through density of the associated routing tile, respectively, which indicates the number of remaining crossing through routes of the tile, l_e is the length of edge e, o_e is the number of overflows of edge e during detailed routing, and a, b, c, d, e, and f are user-defined parameters.

With the above-mentioned cost function for global routes, the Z-pattern router and the framed SP router can calculate the minimum cost route and feed it to the detailed router. If the detailed router succeeds, a procedure for updating the routing resources will be called. Otherwise, the overflow of the failed routing edge is increased.

During the rerouting stage, the global router is the framed SP router. The difference between the trying stage and the rerouting stage lies in the frame size of the searching area. In the rerouting stage, the frame increases little by little to the entire routing area if all previous routing attempts fail. When the frame has increased to the entire routing area and the routing still fails after a predefined number of iterations, we accept the failure of the subnet and start to reroute other subnets.

5　Congestion-driven detailed router

The detailed router is a grid-based router based on the maze-searching algorithm. In order to obtain a high completion rate, we have adopted some heuristics, such as congestion-driven backtrace and point-to-path maze-searching. In order to cure the blindness of the maze routing algorithm, we introduce the notion of grid density to encourage the backtracing of less congested areas. During the routing procedure, we keep a map of the density values of all the detailed routing grids. The density value of a detailed routing grid is calculated as follows:

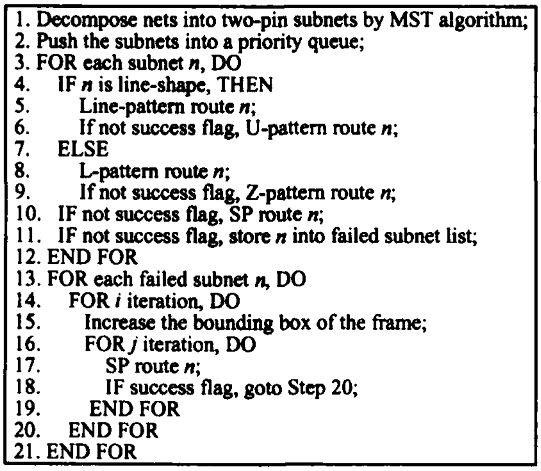
$$d(x, y) = \sum_{x_1 = x - DIS}^{x + DIS} \sum_{y_1 = y - DIS}^{y + DIS} 1/(|x - x_1| + |y - y_1| + 1) \tag{5}$$

where DIS is a user-defined parameter which controls the density area size of a routing grid. The task of this calculation seems time-consuming, but in fact, the initial calculation can be done very quickly, within one second for all the test cases with DIS set at 6. This is because we calculate the grid density from obstacles rather than from the grid itself. For the given test cases, we use over-the-cell routing, which is the same as in Ref. [13]. There are only drawn-up pins which block the routes. If there are n pins drawn up, the time complexity of the calculation will be $O(n \times \mathrm{DIS}^2)$.

During the backtrace stage, we compare the density values of the grids with the same cost and select the one with the least density value. The congestion-driven backtrace may potentially increase the number of vias. But we can add some cost for a turn during the backtrace and balance between the congestion and the number of vias.

6　Overview of the SRF routing flow

Now we have an overview of the SRF routing flow. The main steps of SRF are shown in Fig. 4. First, nets are decomposed into two-pin subnets using the MST algorithm. Then we calculate the priority value for each subnet and push them into a priority queue. Steps 3～12 are carried out during the trying stage and each subnet is attempted to be routed with minimum possible wire length. For each subnet, first we check whether the subnet is of line-shape, that is, whether the routing tiles where the two pins reside are of the same x or y coordinates. If the two pins are in the same routing tile, which means the subnet is a local net, we still regard it as line-shaped and equip the Line-pattern router for this trivial case. If the subnet is of line-shape, then it will first be passed to the Line-pattern router for line routing. If line routing fails, the subnet will be routed by the U-pattern router. The U-pattern router routes subnets in a predefined bounding box which serves as a restriction of the searching space and the total wire length. However, if the subnet is not of line-shape, it is fed to the L-pattern and Z-pattern router.



```
1. Decompose nets into two-pin subnets by MST algorithm;
2. Push the subnets into a priority queue;
3. FOR each subnet n, DO
4.    IF n is line-shape, THEN
5.       Line-pattern route n;
6.       If not success flag, U-pattern route n;
7.    ELSE
8.       L-pattern route n;
9.       If not success flag, Z-pattern route n;
10.   IF not success flag, SP route n;
11.   IF not success flag, store n into failed subnet list;
12. END FOR
13. FOR each failed subnet n, DO
14.    FOR i iteration, DO
15.       Increase the bounding box of the frame;
16.       FOR j iteration, DO
17.          SP route n;
18.          IF success flag, goto Step 20;
19.       END FOR
20.    END FOR
21. END FOR
```

Fig. 4 Main steps of the SRF routing flow

The two routers try to route the subnet one after another if the subnet is not easily routed. If the subnet remains un-routed after the pattern routers, the SP router will be adopted for a further try. During the trying stage, the bounding box of the SP router is determined by the bounding box of the subnet pins. If the SP router fails, we declare failure of the subnet during the trying stage and store it for processing during the rerouting stage.

When all the subnets are processed during the trying stage, steps 13～21 start the rerouting stage and attempt to reroute the failed subnets. During the rerouting stage, the failed subnets are rerouted by the SP router one by one in the same predefined priority order. The only difference in the SP router between the trying stage and the rerouting stage lies in the size of the bounding box for routing. During the rerouting stage, the size of the bounding box increases gradually to the entire routing area. For each size of the bounding box, the SP router is called for several iterations until the routing succeeds. If the bounding box has reached the entire routing area and the subnet remains un-routed after all the iterations, we have to accept the failure and mark the subnet failed.

Since the detailed router is called by the global routers, it does not appear in Fig. 4. Each time the detailed router successfully routes a subnet, a procedure for routing resource update is called. Thus, the resource estimate can be very accurate, which results in better interaction between the global routing and the detailed routing of the SRF routing flow. The accurate routing resource estimation and better interaction between the routing stages can greatly improve the routing completion rate over the tradition approaches.

7 Experimental results

We have implemented the SRF system in C++ on a SUN Enterprise V880 and have tested it on a set of commonly used benchmark circuits. Table 3 illustrates the detailed information of the original test circuits. Since there are abundant routing resources in the original circuits, both MR and SRF can route 100% of all the circuits. To prove the effectiveness of SRF, we only use two routing layers. Note that SRF can support multilayer routing. Since test case Mcc2 is too large to be solved by the two systems in a reasonable time, we do not list the case in the table. In Table 3, the first column shows the names of the circuits, the second one denotes the layout dimensions, the column under " # Subnet " column gives the number of the two-pin subnets, the " # Pin " column gives the total number of pins, and the " # Layer " column shows the number of routing layers in the original circuits.

Table 3 Benchmark circuits

Circuit	Size/μm	# Subnet	# Pin	# Layer
Mcc1	39000 ×45000	1694	3101	4
Primary 1	7552 ×4988	2037	2941	3
Primary 2	10438 ×6468	8197	11226	3
S13207	6590 ×3640	6995	10562	3
S15850	7040 ×3880	8321	12566	3
S38417	11430 ×6180	21035	32210	3
S38584	12940 ×6710	28177	42589	3
S9234	4020 ×2230	2774	4185	3
Struct	4903 ×4904	3551	5717	3
S5378	4330 ×2370	3124	4734	3

Table 4 shows the experimental results of MR and SRF. The "Crit. Ratio" column gives the ratio of the wire length of critical nets to the total wire length after routing by the SRF system. The critical nets are selected from nets whose pins occupy the largest area. Only the number of the nets is selected randomly. It is easy to understand that such nets are always long nets which can easily run into timing violation. "Comp. of Crit." and "Comp. rate" denote the routing completion rate of the critical nets and all nets. The routing completion rate is calculated as the ratio of the routed number of nets to the total number of nets. "WL of Crit." and "WL" give the wire length of the critical nets and all nets respectively. "Runtime" indicates the runtime of both routing systems and "Imp." represents the improvement ratio. From the results, we can see that our SRF routing system can route all 10 test cases with a 100% routing completion rate, while MR can route 100% of only 3 cases which still have abundant routing resources with 2 routing layers. Since these 3 cases still have plenty of routing resource, long nets do not have to detour much to be routed in MR. That is why SRF improves the wire length of the critical nets only a little over MR in these 3 cases. Since long nets often run into timing problems, they require special concern and should be routed with as short a wire length as possible. Thus the performance of the SRF system turns out to be better than MR. At the same time, the total wire lengths of the SRF are no worse than MR even though SRF has routed more nets. Finally, SRF runs faster than MR by up to 382.5%. Thus, as far as routability, performance, and scalability are concerned, SRF proves to be a more effective full-chip routing system.

Table 4 Experimental results of MR and SRF

Circuit	Crit. Ratio	Comp. of Crit.		WL of Crit./10^7			Comp. rate		Runtime/s			WL/10^7	
	/ %	MR	SRF	MR	SRF	Imp./ %	MR	SRF	MR	SRF	Imp./ %	MR	SRF
Mcc1	26.4	0.963	1	755	723	4.4	0.990	1	569.3	497.9	14.3	2880	2850
Primary 1	10.7	1	1	11.3	11.1	1.7	1	1	384.9	205.3	87.4	105	104
Primary 2	17.2	1	1	74.7	73.1	2.1	1	1	2133.7	749.9	184.5	428	426
S13207	30.5	0.960	1	6.16	5.87	4.8	0.996	1	198.0	89.4	121.4	19.9	19.9
S15850	31.7	0.899	1	7.73	7.23	6.8	0.995	1	670.4	253.6	164.4	24.6	24.7
S38417	42.4	0.978	1	23.6	22.6	4.2	0.998	1	606.1	365.8	65.7	54.9	54.1
S38584	40.6	0.986	1	31.9	30.2	5.8	0.997	1	2452.3	966.1	153.8	76.3	75.1
S9234	23.6	0.966	1	1.52	1.43	6.2	0.995	1	35.9	21.9	63.7	6.40	6.35
Struct	38.7	1	1	34.6	34.0	1.8	1	1	547.4	113.5	382.5	88.7	87.8
S5378	17.2	.788	1	1.28	1.17	8.0	0.988	1	72.1	50.8	41.8	8.37	8.54

8 Conclusion

We propose a novel congestion and performance driven full-chip scalable routing framework. Our SRF system features the fast scalable global router, congestion-driven detailed router, and the criticality-driven least flexibility prior net ordering technique. Experimental results show that the SRF system can improve the final routing solution significantly in terms of total wire length of critical nets, routing completion rate, and runtime. As far as timing optimization is concerned, the SRF system proves to be more reasonable and practical.

References

[1] Lee C Y. An algorithm for path connections and its applications. IRE Trans Electron Comput, 1961, 10(3): 346

[2] Soukup J. Fast maze router. Proc IEEE Design Automation Conference, 1978: 100

[3] Hightower D. A solution to line-routing problems on the continuous plane. Proc IEEE Design Automation Conference, 1969: 1

[4] Nair R. A simple yet effective technique for global wiring. IEEE Trans Computer-Aided Design of Integrated Circuits and Systems, 1987, 6(2): 165

[5] McMurchie L, Ebeling C. PathFinder: a negotiation-based performance-driven router for FPGAs. Proc ACM Symposium on Field-Programmable Gate Arrays, 1995: 111

[6] Meixner G, Lauther U. A new global router based on a flow model and linear assignment. Proc IEEE International Conference on Computer-Aided Design, 1990: 44

[7] Albrecht C. Provably good global routing by a new approximation algorithm for multicommodity flow. Proc International Symposium on Physical Design, 2000: 19

[8] Burstein M, Pelavin R. Hierarchical channel router. Proc IEEE Design Automation Conference, 1983: 591

[9] Lin Y L, Hsu Y C, Tsai F S. Hybrid routing. IEEE Trans Computer-Aided Design of Integrated Circuits and Systems,

1987 ,9(2) :151

[10] Heisterman J ,Lengauer T. The efficient solution of integer programs for hierarchical global routing. IEEE Trans Computer-Aided Design of Integrated Circuits and Systems ,1991 , 10(6) :748

[11] Cong J ,Xie M ,Zhang Y. An enhanced multilevel routing system. Proc IEEE International Conference on Computer-Aided Design ,2002 :51

[12] Cong J , Fang J , Zhang Y. Multilevel approach to full-chip gridless routing. Proc IEEE International Conference on Computer-Aided Design ,2001 :396

[13] Lin S P ,Chang Y W. A novel framework for multilevel routing considering routability and performance. Proc IEEE International Conference on Computer-Aided Design ,2002 :44

[14] Ho T Y,Chang Y W ,Chen S J ,et al. A fast crosstalk- and performance-driven multilevel routing system. Proc IEEE International Conference on Computer-Aided Design ,2003 :382

[15] Ho T Y,Chang Y W,Chen S J. Multilevel routing with antenna avoidance. Proc International Symposium on Physical Design ,2004 :34

[16] Ho T Y,Chang C F ,Chang Y W ,et al. Multilevel full-chip routing for the X-based architecture. Proc IEEE Design Automation Conference ,2005 :597

[17] Chen T C ,Chang Y W. Multilevel gridless full-chip routing considering optical proximity correction. Proc IEEE Asia South Pacific Design Automation Conference ,2005 :1160

[18] Kastner R ,Bozorgzadeh E , Sarrafzadeh M. Predictable routing. Proc IEEE International Conference on Computer-Aided Design ,2000 :110

[19] Dijkstra E W. A note on two problems in connection with graphs. Numerische Mathematik ,1959 ,1 :269

考虑拥挤度和性能的全芯片可控布线系统框架*

姚海龙 蔡懿慈 洪先龙† 周 强

(清华大学计算机科学与技术系，北京 100084)

摘要：提出一个全新的全芯片可控布线系统框架 ,同时考虑布线拥挤度和芯片性能. 为了在总体布线和详细布线之间架起桥梁 ,该框架把总体布线和详细布线集成起来 ,交互进行 ,每完成一个线网的布线 ,都及时对布线资源进行更新 ,由此可以得到精确的资源估计结果 ,有利于指导后续总体布线决策. 该系统框架的主要特征包括快速的基于模式的和基于外框约束下最短路算法的总体布线器、基于迷宫算法的拥挤度驱动的详细布线器以及在两个布线器之间很好的交互性. 在该布线系统框架中 ,为了优化电路性能 ,在布线中关键线网被赋予更高的优先级. 同时 ,为了优化不同的布线目标 ,可以采用不同的线网排序策略. 该布线系统框架在一套公用的测试电路上完成测试 ,并与之前提出的多级布线系统框架进行比较 ,实验结果表明 ,文中提出的布线系统框架在电路性能、布通率和运行时间方面都取得了很大改进.

关键词：拥挤度；多级布线；性能；可控布线

EEACC：2570 **CCACC**：7410D

中图分类号：TN47 **文献标识码**：A **文章编号**：0253-4177(2006)07-1201-08

* 国家高技术研究与发展计划资助项目(批准号:2005AA1Z1230)

† 通信作者. Email:hxl-dcs@mail.tsinghua.edu.cn

2005-11-03 收到,2006-02-14 定稿