

# Optimization of Global Signal Networks for Island-Style FPGAs

Ni Minghao<sup>†</sup>, Chan S L, and Liu Zhongli

(*Institute of Semiconductors, Chinese Academy of Sciences, Beijing 100083, China*)

**Abstract:** We present a staggered buffer connection method that provides flexibility for buffer insertion while designing global signal networks using the tile-based FPGA design methodology. An exhaustive algorithm is used to analyze the trade-off between area and speed of the global signal networks for this staggered buffer insertion scheme, and the criterion for determining the design parameters is presented. The comparative analytic result shows that the methods in this paper are proven to be more efficient for FPGAs with a large array size.

**Key words:** circuit design; FPGA; global signal network; optimization; buffer insertion

**EEACC:** 1000

**CLC number:** TP302

**Document code:** A

**Article ID:** 0253-4177(2008)09-1764-06

## 1 Introduction

While implementing digital circuits into an FPGA, there are always signals with high fan-out, such as chip input clock and set/reset signals driving many logic gates and flip-flops. These signals tend to spread over the entire chip and are called global signals. In an FPGA, the dedicated routing circuit used for distributing global signals is called the global signal network (GSN).

The tile-based FPGA design methodology<sup>[1]</sup> can significantly reduce the manpower required to design an FPGA. When designing a GSN in a tile-based FPGA, however, the optimization is very complicated. The challenges come from the chip-wise symmetry required by the tile-based methodology.

In this paper, methods are presented to optimize the GSN circuit under the constraints of the tile-based methodology:

A novel staggered buffer connection method is introduced in detail. This method allows the insertion of a buffer shared by an arbitrary number of GSN tiles. An exhaustive algorithm is adopted to optimize the GSN circuit. Compared with the traditional delay optimization algorithms in Refs. [2, 3], this exhaustive algorithm computes both the delay and area values of the GSN circuit for each possible set of design parameters. Using this algorithm, quantitative analysis of the trade-offs between area and speed of the GSN circuit is performed, and the criterion for determining optimized GSN design parameters is presented. Simulation results show that the GSN circuit with such optimized parameters trades an insignificant delay in-

crease for a remarkable saving in area.

## 2 Background of GSN design

### 2.1 FPGA floorplan

The structure of an island-style FPGA is discussed in Ref. [4]. Taking advantage of the regularity of the FPGA array structure, the tile-based methodology<sup>[1]</sup> is widely used in FPGA design. In Ref. [5], techniques for constructing a tile are introduced. In the tile-based methodology, an FPGA tile consists of the logic block (LB) circuit and its associated generic routing pattern. This tile is replicated in both the horizontal and vertical directions to form a tile array surrounded by IO blocks. Based on the tile-based methodology, a FPGA floorplan improving on that presented in Ref. [4] is adopted in this paper. Figure 1 demonstrates the floorplan for a  $4 \times 4$  FPGA; the cross-shaped rectangular area composed of horizontal and vertical rectangles divides the entire layout into 4 identical  $2 \times 2$  tile arrays. This area contains the programming circuit, which reads in configuration data from external programming hardware and configures the configuration cells in the FPGA. This floorplan can be extended to a large array by increasing the number of tiles in both the horizontal and vertical directions.

### 2.2 Spine-and-ribs network

The structures of the GSNs, taken either from modern commercial FPGAs<sup>[6,7]</sup> or academic research<sup>[8,9]</sup>, are all derived from the spine-and-ribs topology<sup>[8]</sup>. One key reason that this topology is well accepted is that it can be easily integrated into the tile-

<sup>†</sup> Corresponding author. Email: nimi@semi.ac.cn

Received 25 February 2008, revised manuscript received 18 May 2008

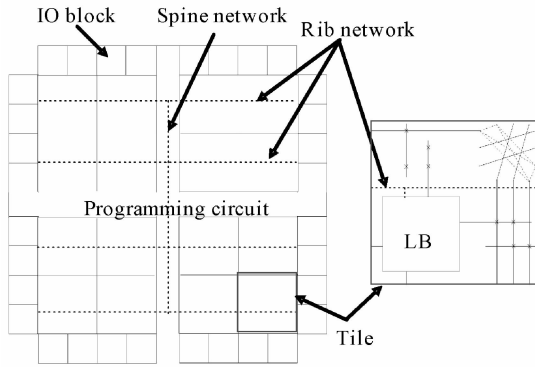


Fig. 1 Improved floorplan for a  $4 \times 4$  tile array

based FPGA. In Fig. 1, the network indicated by the dotted lines illustrates the spine-and-ribs structure. The spine-and-ribs GSN consists of a spine network and several rib sub-networks. A global signal is distributed to each row using the spine network and is connected to the LBs in each row using the rib sub-networks.

Figure 1 also shows how the GSN is partitioned for the convenience of design and abutment. The spine network is folded into the programming circuit layout. The programming circuit and the spine network are both designed and implemented by commercial synthesis and place-and-route tools. It is outside the scope of this paper to discuss these circuits. The rib network is separated into small fragments that are incorporated into the FPGA tiles and can be later put together to reconstruct the rib network while abutting the FPGA tiles.

### 2.3 Inserting buffers into GSN

To minimize the delay of a GSN, buffers need to be inserted. In order to maintain the chip-wise symmetry, the following constraint is imposed on buffer insertion:

Given that all buffers are inserted equidistantly (Reference [3] has proven that it leads to the minimal delay), the length of an FPGA tile should be an integral multiple of the distance between two buffers.

To meet this constraint, Reference [8] proposes the insertion scheme of one buffer per tile. In Ref. [1], no buffer is inserted into GSN. To the best of our knowledge, there is no academic work discussing the optimization of a GSN circuit design for tile-based FPGAs.

## 3 GSNs design optimization

A summary of the parameters describing the circuit of GSNs is shown in Table 1. In this section, the methods for optimizing the design parameters of the

Table 1 Global signal network parameters

Parameter	Parameter description
$S_{buf}$	Size of the buffers inserted into rib networks
$W_{metal}$	Height of the rectangular area reserved in the metal layer for routing rib networks (unit: $\mu m$ )
$W_{spacing}$	Spacing between 2 parallel metal wires of rib networks (unit: $\mu m$ )
$M_{tile}$	Number of tiles that a rib network spans
$L_{tile}$	Length of an FPGA tile (unit: $\mu m$ )
$N$	Number of global signal networks
$D_{buf}$	Density of rib buffers inserted, i.e., number of tiles residing between two connected rib buffers

GSN are introduced.

### 3.1 A novel staggered connection

Even though the scheme in Ref. [8] has a maximum  $D_{buf}$  value of 0, which inserts the least buffers under the constraint described in section 2.3, the simulation result shows that there are still more buffers than needed. Besides area waste, the buffers' intrinsic delay will cause a performance loss. In order to reduce the number of buffers inserted, in this section, a novel staggered connection method is introduced. This method can extend the distance between two connected buffers to an integral multiple of the FPGA tile length.

In most cases, there is more than one GSN in an FPGA chip. The staggered connection method is presented for the design of multiple GSNs in a tile-based FPGA. Figure 2 illustrates the staggered connection of buffers where  $N = 4$  and  $D_{buf} = 1$ . Figure 2 (a) shows an FPGA tile in which the identical fragment from a rib network is inserted. For clarity, only the GSN part is shown. Figure 2 (b) is the rib network constructed by replicating the tile in Fig. 2 (a).

For any given values of  $N$  and  $D_{buf}$ , the staggered connection inserts  $\lceil \frac{N}{D_{buf} + 1} \rceil$  rib buffers into each

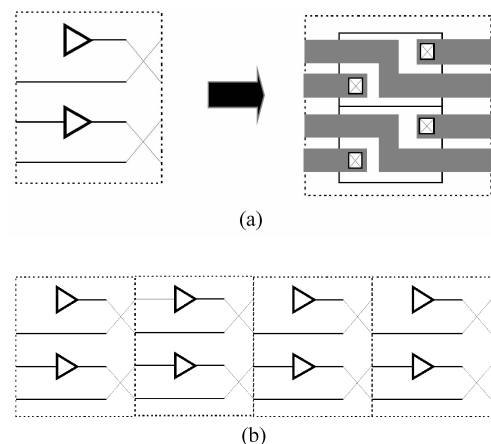


Fig. 2 Staggered connection (a) Fragment of rib network; (b) Rib network

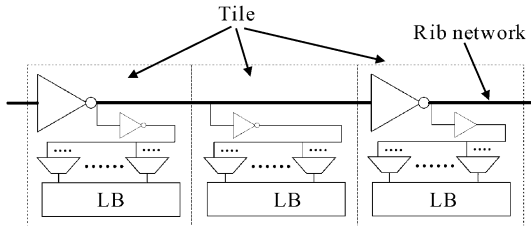


Fig. 3 Structure connecting GSM to LB

tile, where  $\lceil x \rceil$  is the largest integer not greater than  $x$ . A metal wire, connecting to the buffer output of one tile, overpasses the adjacent  $D_{\text{buf}}$  tiles and connects to the input of the buffer in the next tile. Metal wires from all  $N$  rib networks connect the buffers in the same way, except that the connected buffers are staggered from wire to wire to maintain the symmetry required by tile-based methodology.

Compared with the staggered connection technique, the traditional scheme<sup>[8]</sup> that inserts one buffer per tile has the following disadvantages: it fixes  $D_{\text{buf}}$  to 0, and any attempt to insert sparser buffers will violate the uniform structure of the tiles.

While the new scheme presented in this section provides sparser buffer insertion and thereby the possibility for better performance, it also leads to potential area waste. When  $\frac{N}{D_{\text{buf}} + 1}$  is not an integer, there will be one or more unused buffers in the tiles. When computing the consumed area, the wasted area should also be taken into account. So the average area for one rib network should be calculated as follows:

$$\text{Area} = A_{\text{rib\_buf}} \times \left\lceil \frac{N}{D_{\text{buf}} + 1} \right\rceil \times M_{\text{tile}} / N \quad (1)$$

where  $A_{\text{rib\_buf}}$  is the area of one rib buffer.

### 3.2 Determining the GSNs' parameters

In this section, we will first introduce the circuit elements for constructing the RC-tree of the rib network and then the methods for analyzing and determining the GSNs' design parameters.

#### 3.2.1 Calculating delay of rib networks

Figure 3 illustrates a section of the rib network connecting the GSM to LB for  $D_{\text{buf}} = 1$ . This section consists of two rib buffers and a metal wire connecting two adjacent rib buffers. Each buffer of the buffer chain is loaded by the capacitance from the wire and the gates driven. The length of the wire connecting the rib buffers is determined by tile length and  $D_{\text{buf}}$ ; the width of the wire is denoted as  $W_{\text{wire}}$  and calculated as follows:

$$W_{\text{wire}} = W_{\text{metal}} / N - W_{\text{spacing}} \quad (2)$$

The equivalent capacitance and resistance of the wire

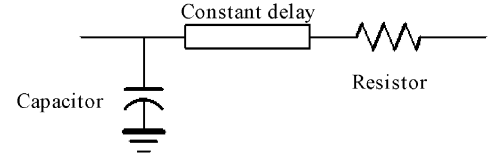


Fig. 4 Equivalent circuit of buffer

can then be computed using the distributed RC model depicted in Ref. [10]. As shown in Fig. 4, the model of buffers in this paper is composed of resistors, capacitors, and constant-delay elements (see Ref. [1] for details). The values of the equivalent resistance, capacitance, and buffer intrinsic delay for each possible buffer size are manually calculated in advance by circuit simulations using HSPICE. The buffer model together with the wire capacitance and resistance are used to build an equivalent RC-tree for the rib network. It then computes the Elmore delay of the rib network.

#### 3.2.2 Determining optimal rib buffer size and density

From the previous discussion, we find that, in tile-based methodology, buffers can only be inserted at discrete points. The limited flexibility of the buffer locations reduces the complexity of the optimization process and makes it feasible to search the entire parameter space for an optimized solution. In this section, this method of exhaustion is adopted to analyze the GSN circuit for each possible design parameter.

Using this method, we have computed the delay values of a rib network spanning 10 tiles for all possible  $D_{\text{buf}}$  and  $S_{\text{buf}}$  (using a 0.5  $\mu\text{m}$  SOI technology, the relevant parameters are listed in Table 3), and plotted the results in Fig. 5(a). The tag number on each curve is the value of  $D_{\text{buf}}$  for which the curve is computed. Each curve has a point of minimal delay. These points are connected in the ascending order of  $D_{\text{buf}}$  and shown as a dotted line. The minimal delay of 3.127ns can be easily identified as  $S_{\text{buf}} = 14$  and  $D_{\text{buf}} = 4$ .

Given that we can always trade area for speed, and vice versa, it makes sense to combine these two factors into one curve to see where the best trade-off occurs. In Fig. 5(a), the minimal delay point of each curve divides the curve into two segments. Within the first segment (left of the minimal delay point), increasing  $S_{\text{buf}}$  leads to a delay reduction. We denote the maximum and minimum values of Delay and Area of the first segment as  $D_{\text{max}}$ ,  $D_{\text{min}}$ ,  $A_{\text{max}}$ ,  $A_{\text{min}}$ , respectively. In this segment, the average delay reduction gained by increasing area is  $\frac{D_{\text{max}} - D_{\text{min}}}{A_{\text{max}} - A_{\text{min}}}$ . So, we use Eq. (3) to calculate the cost values for a rib network:

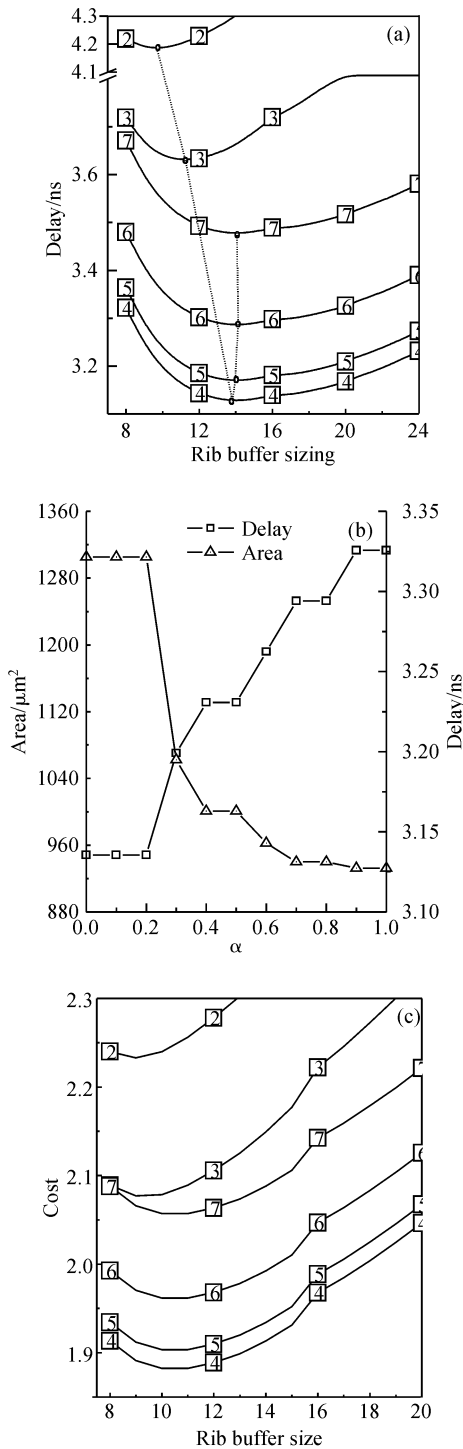


Fig.5 Quantitative analysis results (a) Delay of rib network versus rib buffer size; (b) Delay and area obtained at minimum cost points; (c) Cost versus rib buffer size

$$\text{Cost} = \alpha \times \text{Delay} + (1 - \alpha) \times \frac{D_{\max} - D_{\min}}{A_{\max} - A_{\min}} \times \text{Area} \quad (3)$$

where Area is the silicon area of one rib network calculated using Eq. (1), Delay is the delay of the rib network.  $\alpha$  is a parameter that controls the weights of Delay and Area in the trade-off. The cost value is used as the metric for the suitability of the size and density

Table 2 Best trade-off points at various values of  $N$

$N$	Min. cost (best trade-off points)			Min. delay (3.127ns)	Delay increasing	Area saving	
	$D_{\text{buf}}$	$S_{\text{buf}}$	Area / $\mu\text{m}^2$	Delay /ns			Area / $\mu\text{m}^2$
1	4	10	6688	3.195	8208	2.17%	18.5%
2	4	10	3344	3.195	4101	2.17%	18.5%
3	4	10	2229	3.195	2736	2.17%	18.5%
4	4	10	1672	3.195	2052	2.17%	18.5%
5	4	10	1337	3.195	1641	2.17%	18.5%
6	5	11	1178	3.204	2736	2.46%	56.9%
7	6	14	1172	3.285	2345	5.05%	50.0%
8	4	10	1672	3.195	2052	2.17%	18.5%

of rib buffers.

Finding a good choice of  $\alpha$  is very important. We analyzed how the values of Delay and Area are affected by  $\alpha$  at all the minimal cost points. Figure 5 (b) shows such effect for the rib network with  $D_{\text{buf}} = 4$  and  $N = 4$ . The figure indicates that an  $\alpha$  value from 0.3 to 0.5 is acceptable. For different rib network structures, the results are approximately the same.

The  $(S_{\text{buf}}, D_{\text{buf}})$  set of values with the minimal cost is regarded as the optimal buffer insertion scheme for the rib networks. Like in Fig. 5 (a), we plotted the cost value for  $N = 4$  in Fig. 5 (c) ( $\alpha$  is 0.5). The best trade-off point for the given  $N$  can then be identified from the plot as  $S_{\text{buf}} = 10$  and  $D_{\text{buf}} = 4$ . For various values of  $N$ , the best trade-off points are listed in Table 2. The area and delay of each of these points are compared with those from the corresponding minimum delay points. The results show that an average delay increase of 2.56% is traded for a 26.38% savings in area when  $\alpha$  is 0.5. Table 2 shows that when  $N$  equals 6 or 7, the area saved is much higher. This result comes from the fact that at the minimum delay point,  $D_{\text{buf}}$  is 4 and two buffers are inserted into each tile, while at the best trade-off point only one is inserted by changing the  $D_{\text{buf}}$  value. The criteria demonstrate its ability to automatically select a suitable  $D_{\text{buf}}$  value to avoid significant area waste inherent in the staggered buffer connection technique.

Although the optimization process in this section assumes a particular FPGA architecture and a  $0.5\mu\text{m}$  SOI technology, the entire process can be used for other FPGA architectures and manufacturing technologies. As such, the entire process is automated by a software tool. This tool reads in an architecture specification file describing all needed parameters for GSNs optimization, and outputs the netlist and layout of the rib network fragments.

### 3.2.3 Time complexity of the exhaustive algorithm

Recall that the RC-tree representing a rib network is constructed by connecting the sub-circuits in each FPGA tile. Let  $p$  denote the average number of

Table 3 Key parameters of GSNs

Parameter	Value	Parameter	Value
$W_{\text{metal}}$	$18\mu\text{m}$	$W_{\text{spacing}}$	$3\mu\text{m}$
$L_{\text{tile}}$	$1900\mu\text{m}$	$\alpha$	0.5
Range of buffer size	From 1 to 30	$N$	4

the electrical nodes in a rib network sub-circuit, the number of nodes in a RC-tree can be computed as  $pM_{\text{tile}}$ . For the time complexity, we only consider the operation of visiting RC-tree nodes because the time-consuming multiplication is performed then. Calculating RC-tree delay requires traversing the RC-tree twice and can be obtained with  $2pM_{\text{tile}}$  complexity. An exhaustive search process performs the operation of delay calculation for every possible  $D_{\text{buf}}$  and  $S_{\text{buf}}$  value. Consequently, the overall time complexity is  $2(D_{\text{buf,max}} - D_{\text{buf,min}} + 1)(S_{\text{buf,max}} - S_{\text{buf,min}} + 1)pM_{\text{tile}}$ . In this paper,  $S_{\text{buf}}$  is between 8x to 30x and  $D_{\text{buf}}$  is from 0 to  $M_{\text{tile}} - 1$ . So an exhaustive search process can be finished by visiting RC-tree nodes  $46pM_{\text{tile}}^2$  times and has a computing complexity of  $O(M_{\text{tile}}^2)$ .

## 4 Experimental and simulation results

In this paper, the targeted manufacturing technology is  $0.5\mu\text{m}$  SOI technology. Some key parameters needed for describing GSNs are listed in Table 3.

The optimal design of GSNs is created using the process described in section 4. The delay value obtained is compared with those obtained using the buffer insertion schemes in Refs. [1,8], which are also used for GSN design under the tile-based constraints. The results are plotted in Fig. 6. Within a small chip size ( $M_{\text{tile}} < 14$ ), a rib network without any buffer is the best design because it has minimal delay and consumes almost no silicon area. However, since no buffer is inserted in this scheme, the delay

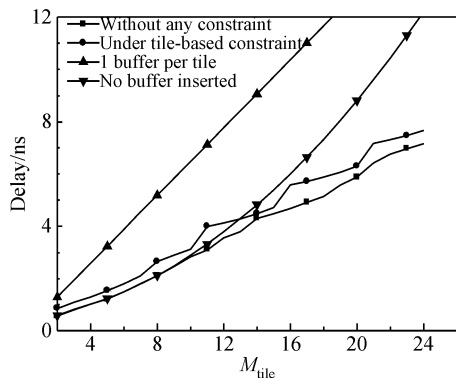


Fig. 6 Comparison results

value increases quadratically with its length. When  $M_{\text{tile}} \geq 14$ , the method we proposed can achieve a better performance. Among all the considered insertion schemes, inserting one buffer per tile is the most inefficient.

Moreover, the minimal delay that the rib network can obtain without any additional constraint is also computed using the method in Ref. [3] and plotted in Fig. 6. Compared with this method, the delay from our scheme has an average increase of 17.2%. This is due to the constraints imposed by the tile-based methodology.

## 5 Conclusion

In this paper, we proposed a novel staggered buffer connection method to provide flexibility for buffer insertion while designing GSNs for tile-based FPGA. An exhaustive algorithm is adopted to analyze the effect of all design parameters. The criterion and method for determining optimal trade-off points for design parameters are also presented. Experimental results based on a  $0.5\mu\text{m}$  SOI technology have shown that for FPGAs with a comparatively large array size, the optimization process demonstrates better efficiency.

## References

- [1] Kuon I C. Automated FPGA design, verification and layout. EE Department of University of Toronto; Toronto, 2003
- [2] Van Ginneken L P P P. Buffer placement in distributed RC-tree networks for minimal Elmore delay. Proc Int Symp Circuits and Systems, 1990
- [3] Wang Yibo, Cai Yici, Hong Xianlong. An interconnect optimization algorithm in SOC layout design. Chinese Journal of Semiconductors, 2003, 24(5): 550 (in Chinese) [王一博, 蔡懿慈, 洪先龙. SOC 版图设计中的互连优化算法. 半导体学报, 2003, 24(5): 550]
- [4] Betz V, Rose J, Marquardt A. Architecture and CAD for deep-submicron FPGAs. Kluwer Academic Publishers, 1999
- [5] Padalia K, Fung R, Bourgeault M. Automatic transistor and physical design of FPGA tiles from an architectural specification. Proceedings of the FPGA, Monterey, California, US, 2003
- [6] Xilinx. Vertex-II Pro and Vertex-II Pro X platform FPGAs: Complete Data Sheet, 2005
- [7] Xilinx. Xilinx Spartan-3 FPGA Family Complete Data Sheet, 2006
- [8] Lamoureux J, Wilton S. FPGA clock network architecture: flexibility vs area and power. Proceedings of the FPGA, Monterey, Ca, US; ACM, 2006
- [9] Andrews B W. Patent: Global signal distribution with reduced routing tracks in an FPGA. US: Lucent Technologies Inc, 2000
- [10] Rabaey J M, Chandrakasan A, Nikolic B. Digital integrated circuits: a design perspective. 2nd ed. Prentice Hall, 2003

## 岛式 FPGA 全局信号网络的优化设计

倪明浩<sup>†</sup> 陈陵都 刘忠立

(中国科学院半导体研究所, 北京 100083)

**摘要:** 针对使用拼接单元块设计方法的岛式 FPGA, 介绍了一种交叉连接的方法, 可以为其全局信号网络的缓冲器插入提供可变性. 对于采用此方法设计的全局信号网络, 文中的穷举法定量分析了其面积和性能之间相互影响的关系, 并提出了量化的标准, 用于选择设计参数使面积和性能达到均衡. 通过对比常用方法和文中方法所得到的全局信号网络的性能, 证明文中方法在较大的 FPGA 芯片中能够得到更优的结果.

**关键词:** 电路设计; FPGA; 全局信号网络; 优化; 缓冲器插入

**EEACC:** 1000

**中图分类号:** TP302      **文献标识码:** A      **文章编号:** 0253-4177(2008)09-1764-06

<sup>†</sup> 通信作者, Email: nimi@semi.ac.cn

2008-02-25 收到, 2008-05-18 定稿