

Channel Segmentation Design for Symmetrical FPGAs*

Wai-Kei Mak and D. F. Wong
wkmak@cs.utexas.edu, wong@cs.utexas.edu
Department of Computer Sciences
University of Texas at Austin
Austin, TX 78712

December 6, 1996

Abstract— The channel segmentation design problem for symmetrical FPGAs is the problem of designing segmented tracks in the interconnection channels to provide good net routability and delay performance at the same time. In this paper, we show how to decompose the problem into the segmentation design problems of the vertical and horizontal channels by a statistical analysis of the net distribution on a symmetrical FPGA. And we propose an effective approach for segmented channel design when the allowed number of tracks in a channel is fixed and limited.

1 Introduction

Field Programmable Gate Arrays (FPGAs) are logic devices that can be programmed by the users to implement different circuit designs. They are widely used for system prototyping and logic emulation because of their short production time and low prototyping cost.

Figure 1 shows the basic symmetrical FPGA architecture. A symmetrical FPGA chip consists of a two-dimensional array of logic cells. The routing channels between adjacent rows and between adjacent columns of logic cells provide paths to realize the inter-cell connections required for implementing a given circuit design. A routing channel contains a set of tracks. At the intersection of a vertical channel and a horizontal channel, there are some *crossing* switches, each of which can be programmed to establish a link between a vertical and a horizontal track [1, 2, 3]. Each track on the chip is divided into a number of segments. And there is a *continuing* switch between every two neighbouring segments on a track. By programming these switches, consecutive segments can be linked together to realize longer paths.

In the early FPGAs, the segments are of unit length, as a result each track is highly populated with switches. Since the switches have high resistance and capacitance, they introduce significant

*This work was partially supported by a grant from Lucent Technologies.

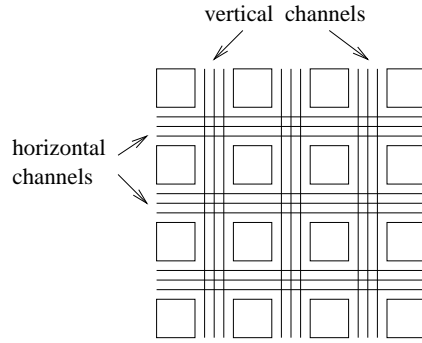


Figure 1: Basic symmetrical FPGA architecture.

delay even for routing a net of moderate length. Though we may reduce the number of switches on the tracks to reduce the delay, this will also reduce routability. This tradeoff between delay performance and routability presents a segmentation design problem. And in more recent FPGAs, tracks making up of longer segments are provided in addition to the unit-length segmented tracks.

In this paper, we address the channel segmentation design problem for the symmetrical FPGAs. We consider using a set of different types of tracks and determine how many tracks in each type should be used in a channel to provide high routability and good delay performance when the total number of tracks is already fixed. In particular, we adopt a *regular segmentation model* where each track is divided into segments of equal length. See Figure 2 for an example channel of the regular segmentation model with three types of tracks.

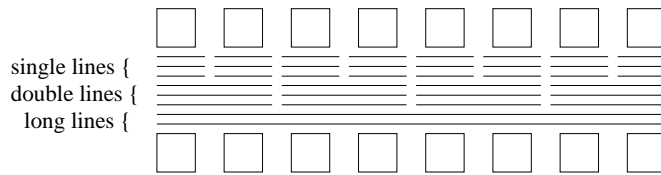


Figure 2: Regular segmentation model.

The rest of the paper is organized as follows. In section 2, we show how to reduce the symmetrical FPGA channel segmentation design problem into the design problems of vertical and horizontal channels when the issue of routing delay is taken into account. In section 3, we propose a new approach for channel segmentation design using the regular segmentation model. Finally, we present the experimental results and the conclusions in section 4 and section 5, respectively.

2 Channel Segmentation Design for Symmetrical FPGAs

In this section, we consider the channel segmentation design problem for symmetrical FPGAs. We show how to decompose it to the segmentation design problems of vertical and horizontal channels

based on a statistical analysis of the distribution of the net. As was observed in [4], segmentation design for a symmetrical FPGA can be obtained by constructing channel segmentations for the horizontal and vertical channels independently by using some channel segmentation design algorithm for a single channel [5, 7, 6]. Here we show in details how this can be done. We present a decomposition which takes into account the issue of delay to ensure good overall delay performance.

Following [8], we model a symmetrical FPGA chip as a two-dimensional grid where the grid-lines correspond to the channels and each grid-point corresponds to the intersection of a horizontal and a vertical channel. We assume that nets in a symmetrical FPGA chip are routed in a minimum distance fashion. Most routing algorithms attempt to accomplish this because it can minimize net delay and channel congestion. Moreover, to guarantee that the delay for an inter-cell connection is not too large, we should also limit the number of turns it is allowed to make because every turn makes use of one crossing switch which incurs significant delay. In the following, we assume that each connection can make at most two turns (but the same method also applies for any other fixed number of turns allowed). This implies that each connection can consist of at most three pieces (see Figure 3). Finally, we should limit the number of switches used by each piece.

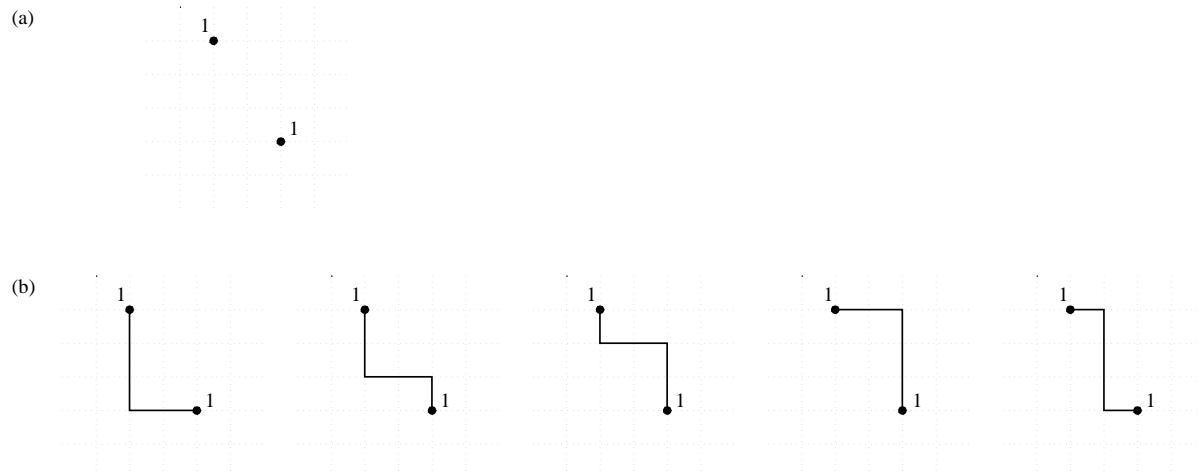


Figure 3: (a) A net. (b) Five desirable ways to connect the net.

Assuming the nets are routed in a minimum distance fashion and can make at most two turns, the number of routes between any two grid-points is equal to the Manhattan distance between them (provided that they are in different rows and different columns). For example, the number of ways to route net 1 in Figure 3(a) is five. We will assume that these routes are equally likely to be taken for routing a net between the two grid-points.

We can perform a statistical analysis to determine the length distributions of the vertical pieces and of the horizontal pieces which make up the final connections. This will allow us to tailor the vertical channel segmentation pattern for routing the vertical pieces and tailor the horizontal channel segmentation pattern for routing the horizontal pieces. Hence we may reduce the channel

segmentation design problem of a symmetrical FPGA to the channel segmentation design problem of the vertical and horizontal channels.

We will first demonstrate how to do the analysis using a very simple example. In this example, we assume that there are only two types of nets to simplify the discussion. The two types of nets are shown in Figure 4. The second terminal of a type 1 net is two rows below and two columns to the right of its first terminal. The second terminal of a type 2 net is two rows below and one column to the left of its first terminal. There are four ways to route a type 1 net (Figure 5(a)) and each has a probability of $\frac{1}{4}$ to be taken. There are three ways to route a type 2 net (Figure 5(b)) and each has a probability of $\frac{1}{3}$ to be taken. Suppose the proportions of type 1 and type 2 nets are 40% and 60%, respectively.

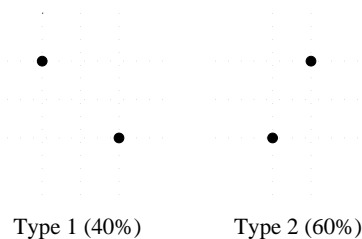


Figure 4: Net distribution.

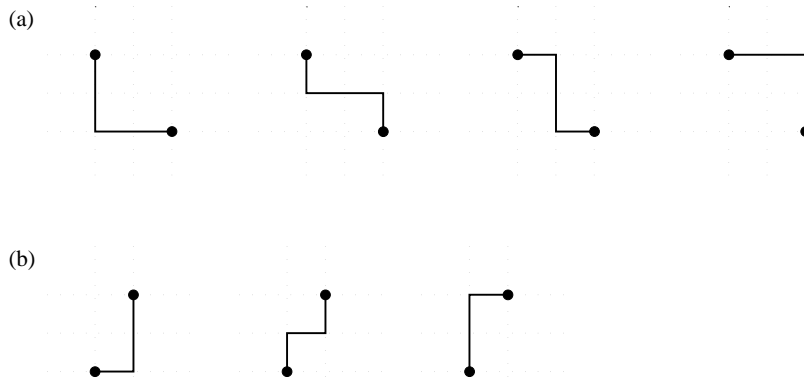


Figure 5: Desirable ways of connection.

To find the distribution of the length of the vertical pieces, we calculate the expected numbers of unit-length and double-length vertical pieces a type 1 net contributes, and the expected numbers of unit-length and double-length pieces a type 2 net contributes.

Expected number of unit-length vertical pieces from a type 1 net is $0 \times \frac{1}{4} + 2 \times \frac{1}{4} + 0 \times \frac{1}{4} + 0 \times \frac{1}{4} = \frac{1}{2}$.

Expected number of unit-length vertical pieces from a type 2 net is $0 \times \frac{1}{3} + 2 \times \frac{1}{3} + 0 \times \frac{1}{3} = \frac{2}{3}$.

Expected number of double-length vertical pieces from a type 1 net is $1 \times \frac{1}{4} + 0 \times \frac{1}{4} + 1 \times \frac{1}{4} + 1 \times \frac{1}{4} = \frac{3}{4}$.

Expected number of double-length vertical pieces from a type 2 net is $1 \times \frac{1}{3} + 0 \times \frac{1}{3} + 1 \times \frac{1}{3} = \frac{2}{3}$.

Hence the expected ratio of the number of unit-length vertical pieces to the number of double-length vertical pieces is $(40\% \times \frac{1}{2} + 60\% \times \frac{2}{3}) : (40\% \times \frac{3}{4} + 60\% \times \frac{2}{3}) = 6 : 7$. So the expected percentages of vertical pieces having unit-length and double-length are 46% and 54%, respectively. In the same way, we can compute the expected percentages of horizontal pieces having unit-length and double-length.

The procedure for finding the distribution of the length of the vertical (horizontal) pieces is described formally below.

We can divide the nets on a $N \times N$ grid into four major classes according to the relative positions of its terminals as shown in Figure 6. A net whose terminals lie in the same column belongs to class *A*. A net whose terminals lie in the same row belongs to class *B*. A net that has a terminal located to the right and below its other terminal belongs to class *C*. A net that has a terminal located to the right and above its other terminal belongs to class *D*. We can further divide each major class of nets into different subclasses according to the vertical and horizontal distances between the net terminals. There are $N - 1$ subclasses for class *A* because the vertical distance between two distinct grid-points in a column can range from 1 to $N - 1$. Similarly, there are $N - 1$ subclasses for class *B*. There are $(N - 1) \cdot (N - 1)$ subclasses for class *C* because the vertical distance can range from 1 to $N - 1$ and the horizontal distance also can range from 1 to $N - 1$. Similarly, there are $(N - 1) \cdot (N - 1)$ subclasses for class *D*. So there are $2(N - 1)N$ different types of nets in all.

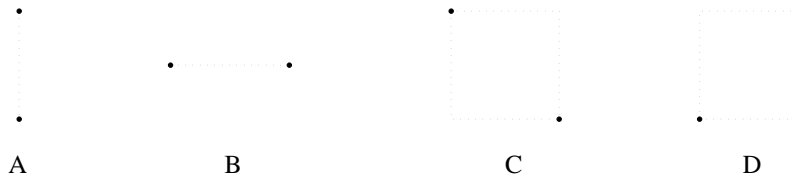


Figure 6: Relative terminal positions for the four classes of nets.

For the k th ($k = 1, 2, \dots, 2(N - 1)N$) type of net, we enumerate all the possible ways to route it in a minimum distance fashion with at most two turns; we denote the number of such routes by r_k . Let the number of vertical pieces of length l in the i th ($i = 1, 2, \dots, r_k$) route be $v_{ki}(l)$. Since all the r_k routes are equally likely to be taken to route a type k net, the expected number of vertical pieces of length l contributed by a type k net is $v_k(l) = \sum_{i=1}^{r_k} v_{ki}(l) \cdot \frac{1}{r_k}$. And the expected number of vertical pieces of length l when all the connections of the nets on the grid are realized is $\sum_{k=1}^{2(N-1)N} n_k v_k(l)$ where n_k is the number of nets of type k . The distribution of the length of the vertical pieces can then be computed. Thus the distribution of the length of the horizontal pieces can also be found in a similar way.

Knowing the length distributions of both the vertical and horizontal pieces means that we can independently tailor the vertical and horizontal channel segmentation patterns for routing the vertical and horizontal pieces. Next, we will address the channel segmentation design problem of individual channels.

3 Channel Segmentation Design

In this section, we introduce a new approach for channel segmentation design.

To route a net, there will be greater flexibility if we allow a few adjacent segments in the same track to be joined end-to-end by continuing switches. But in order to upper bound the routing delay of a net in a channel, we need to limit the number of continuing switches, and hence the number of segments, that the net passes through. A K -segment channel routing is a routing that assigns each net to a track such that no segment is occupied by more than one net and each net occupies at most K segments. For example, a 3-segment channel routing exists for the channel routing instance in Figure 7 (we may route net 1 on the lower track and net 2 on the upper track to obtain a 3-segment routing), but there is no 1-segment or 2-segment channel routing for it.

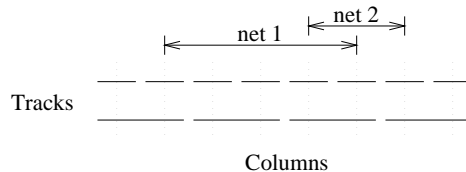


Figure 7: Two nets in a segmented channel

Previous works [5, 6, 7] on routing channel design are mostly based on the segmentation design procedure in [5] for 1-segment channel routing. The major contribution in [5] is that they proved that asymptotically a segmented routing channel can be nearly as efficient as a freely customized routing channel without the restriction of using fixed segments of pre-determined lengths. In particular, they showed how to design a segmented channel with width at most a few times that of a freely customized channel to achieve a high probability of routing completion when the channel length is large. But this is not good enough in reality because the channel width is limited in size due to area constraint.

In [7], a few shortcomings of the procedure in [5] was described and a more sophisticated procedure was suggested to overcome the shortcomings. However, using their procedure, each track has a different segmentation pattern and the segments in the same track have different lengths. This is different from the commercially available symmetrical FPGAs where the types of tracks are not too many and all segments from the same type of tracks have equal length. So in this paper, we propose a new approach for channel segmentation design based on statistical analysis when the number of types of tracks is fixed and each track is divided into segments of equal length.

The main idea of our algorithm is as follows. If we are given a set of different types of tracks, the number of tracks of each type assigned to a channel of fixed width should be proportional to the utility of the type where a high utility means that the segments of a track are expected to be highly demanded by the nets to complete the routing. We will next describe how to measure the track type utility.

For K -segment routing, we want to route each net using no more than K consecutive segments. And we also want to avoid excessive waste of segments so that we can route more nets simultaneously. For example, we will not prefer to route net i on a type 1 track in Figure 8 because the net only spans a small portion of the track but if we route the net using this track, the whole track will become unusable for routing any other net. Instead, we will prefer routing it on a type 2 track.

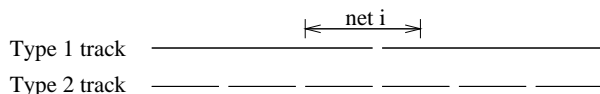


Figure 8: Track selection for routing a net.

So a good routing criterion for a net is that it spans no more than K segments of a track and the total length of the segments occupied by it is no more than α times its length where α is some small constant greater than 1.

Given the net distribution, the *demand* for a particular segment of a track is defined as the expected proportion of nets that can be routed using it under the above criterion when we put the track in a channel. For example, suppose we take $K = 3$ and $\alpha = 1.5$, and we consider a routing channel with two types of tracks and a net distribution as shown in Figure 9. To route an A net, there are two good choices, either using the 1st segment (counting from the left) of a type 1 track, or the 1st to the 3rd segments of a type 2 track. But there is only one good choice for routing a B net, which is using the last segment of a type 2 track. (Though a B net can fit in the last segment of a type 1 track, it is not a good candidate because the length of the segment is more than $\alpha (= 1.5)$ times the length of a B net.) Similarly, there is only one good choice for routing a C net, which uses the 3rd and the 4th segments of a type 2 track. And there are two good choices to route a D net, either using the last segment of a type 1 track, or the last two segments of a type 2 track. Since the 1st segment of a type 1 track will only be used to accommodate an A net under the forementioned good routing criterion and the percentage of A nets is 20%, the demand for the 1st segment of a type 1 track is 20%. As the 2nd segment of a type 1 track will only be used to accommodate a D net, the demand for the the 2nd segment of a type 1 track is 25%. And the demands for the 1st to the 6th segments of a type 2 track are 20%, 20%, 20% + 25%, 25%, 25%, and 25% + 30%, respectively. Note that the demand for the 3rd segment of a type 2 track is 20% + 25% because it can be used to accommodate an A net or a C net under the good routing criterion.

The *utility* of a track type is defined as the average demand of the segments of a track of that

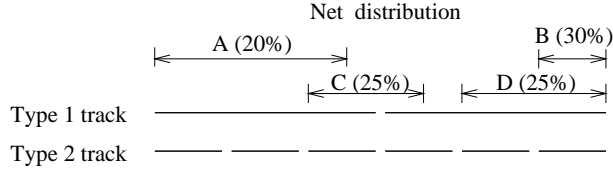


Figure 9: Net distribution.

type. If a track type has a high utility, we should have more tracks of that type. So the number of tracks of each type we assign to a channel of fixed width is proportional to the utility of the type. The complete channel design procedure is given below.

Channel Design for K -segment Routing

Inputs: L = channel length, T = total number of tracks,
 U = number of types of tracks, l_u = segment length of a type u track,
 α = good routing criterion constant,
 $h(x, l)$ = probability that a net is of length l and its left-end is at column x .

Output: $number(u)$ = number of tracks of type u to use.

```

/* Initialize demand(u, j), the demand of the jth segment of a type u track to zero for all u, j */
for x = 0 to L do
  for l = 0 to L - x do
    for u = 1 to U do
      if a net from columns x to x + l spans segments s to s + i - 1 of a type u track
        and i ≤ K and i · l_u ≤ α · l then
          for j = 0 to i - 1 do
            demand(u, s + j) = demand(u, s + j) + h(x, l);
          rof;
        fi;
      rof;
    rof;
  rof;
  rof;
cumulative_sum(0) = 0;
for u = 1 to U do
  utility(u) = (∑_{i=1}^{⌊L/l_u⌋} demand(u, i)) / ⌊L/l_u⌋;
  cumulative_sum(u) = cumulative_sum(u - 1) + utility(u);
rof;
for u = 1 to U do
  number(u) = ⌊ (cumulative_sum(u) / cumulative_sum(U)) · T - (cumulative_sum(u-1) / cumulative_sum(U)) · T ⌋;
rof;

```

Here we suggest one possible way to select the segment lengths for the different types of tracks. The shortest segment lengths we use in a channel can be one, two, and four as are now commonly used in the commercial symmetrical FPGAs. The length of the other types of tracks to

use depend on the channel length L and the value of K where K -segment routing is desired. The longest segment length needed is $\lceil L/K \rceil$, because this guarantees that any net will be routable on such longest segment track using K -segment channel routing. If we want to use $3 + m$ types of tracks, the segment lengths of the different types of tracks we pick will be $1, 2, 4, 4r, 4r^2, \dots, 4r^m$ where $r = (\frac{\lceil L/K \rceil}{4})^{\frac{1}{m}}$. The value of r should be less than the value of K so that no matter what the length of a net is, it will be routable on at least one type of tracks satisfying the good routing criterion.

4 Experimental Results

We used our channel segmentation procedure to design routing channels for different net length distributions assuming the net left-end points follow a uniform distribution. (This choice of distribution for the net left-end point is very close to reality as confirmed by empirical studies [6].) Moreover, we generated different routing problem instances and determined the success rate of routing on the channels using the multi-segment routing algorithm from Zhu *et al.* [7].

The following notations are used:

L : Channel length;

T : Number of tracks in a channel;

K : Maximum number of segments for routing a net;

D : Maximum number of net terminals occupying a column;

$h(x, l)$: Probability that a net is of length l and its left-end is at column x ;

$f(l)$: Probability that a net is of length l .

First we set the channel length L to 20 and the number of tracks T to 18. We noted the routability of the channels produced by our channel segmentation design procedure for nine different distributions of net length (see Table 1). Distributions $b_i, i = 1, 2, \dots, 6$ are defined as follows. If $f(l) = (p_1, p_2, p_3, p_4, p_5)$, then the probability that a net has length within the interval $(0.2(j-1)L, 0.2jL]$ is equal to $p_j / (\sum_{1 \leq k \leq 5} p_k)$. “Ge”, “No”, and “Po” are geometric, normal, and Poisson distributions, respectively. For each distribution, 300 routing problem instances were generated, and then the function $h(x, l)$ was extracted. The net distribution $h(x, l)$, along with the values of L, T, D , and K were used as the inputs to our segmentation design procedure. And we set the value of α in the good routing criterion to be 1.5, i.e., a routing of a net is considered good if the total length of the segments occupied by it is no more than 1.5 its length.

Then we tried to route the 300 randomly generated problem instances of each distribution on the segmentation channel designed for that distribution. The percentage of successful routing for the instances with the same channel density in a distribution was noted. As in [7], we define the *threshold density* d_T as the smallest channel density d such that less than 90% of the instances with density d in the distribution are successfully routed.

We used $L = 20, T = 18, D = 6$, and $K = 2$ for the experimental results reported in Table 1.

The performance of the channels produced by our procedure is compared with that obtained by the procedure in [7].

Table 1: Experimental results.

	$f(l)$	[7]		Ours	
		d_T	d_T/T	d_T	d_T/T
b1	(1, 1, 1, 1, 1)	15	0.83	17	0.94
b2	(1, .8, .5, .3, .1)	14	0.78	15	0.83
b3	(1, .5, .3, .1, 0)	13	0.72	14	0.78
b4	(1, .5, .3, .5, 1)	13	0.72	16	0.89
b5	(.2, .5, 1, .5, .2)	16	0.89	15	0.83
b6	(1, .2, .1, 0, 0)	11	0.61	14	0.78
Ge	$\gamma^l, \gamma = 0.7$	12	0.67	13	0.72
No	$\mu = 4, \sigma^2 = 10$	16	0.89	15	0.83
Po	$\lambda^l e^{-\lambda}/l!, \lambda = 3.0$	13	0.72	15	0.83
		<i>mean = 0.76</i>		<i>mean = 0.83</i>	

And we repeated the experiments using some longer and wider channels. We used $L = 50$, $T = 24$, $D = 8$, and $K = 3$ and got the results are shown in Table 2.

From Tables 1 and 2, we can see that the threshold densities of our channels are better in most cases even though our channels are generated under a more restricted model which requires the segments in the same track to have the same length. This demonstrates that the regular segmentation model is a good choice in segmented channel design, and our approach for determining a good mix of tracks under the regular segmentation model is very effective in designing good vertical and horizontal channels in symmetrical FPGAs.

5 Conclusions

We have shown that channel segmentation design for the symmetrical FPGA architecture with good delay performance and routability can be obtained by a decomposition into the segmentation designs of the vertical and horizontal channels. And we proposed a new effective approach for channel segmentation design when the number of tracks in a channel is fixed and limited, which can be directly applied to the design of the vertical and horizontal channels in symmetrical FPGAs.

References

- [1] AT&T Microelectronics, *Optimized Reconfigurable Cell Array (ORCA) Series Field-*

Table 2: Experimental results.

	$f(l)$	[7]		Ours	
		d_T	d_T/T	d_T	d_T/T
b1	(1, 1, 1, 1, 1)	21	0.88	22	0.92
b2	(1, .8, .5, .3, .1)	17	0.71	21	0.88
b3	(1, .5, .3, .1, 0)	18	0.75	21	0.88
b4	(1, .5, .3, .5, 1)	19	0.79	22	0.92
b5	(.2, .5, 1, .5, .2)	22	0.92	22	0.92
b6	(1, .2, .1, 0, 0)	17	0.71	20	0.83
Ge	$\gamma^l, \gamma = 0.875$	20	0.83	20	0.83
No	$\mu = 8, \sigma^2 = 15$	21	0.88	20	0.83
Po	$\lambda^l e^{-\lambda}/l!, \lambda = 8.0$	19	0.79	19	0.79
		<i>mean = 0.81</i>		<i>mean = 0.87</i>	

Programmable Gate Arrays, Advance Data Sheet, February, 1993.

- [2] H.C. Hsieh, et. al., “Third-Generation Architecture Boosts Speed and Density of Field-Programmable Gate Arrays”, *Proc. of CICC*, pp.31.2.1-31.2.7, 1990.
- [3] Xilinx Inc., *XC 4000 Logic Cell Array Family*, Technical Data, 1990.
- [4] K. Zhu, D.F. Wong, and Y.W. Chang, “Switch Module Design with Application to Two-Dimensional Segmentation Design”, *Proc. of ICCAD*, pp.480-485, 1993.
- [5] A. El Gamal, J. Greene, and V. Roychowdhury, “Segmented Channel Routing is Nearly as Efficient as Channel Routing (and Just as Hard)”, *Advanced Research in VLSI*, pp. 193-221, 1991.
- [6] M. Pedram, B.S. Nobandegani, and B.T. Preas, “Architecture and Routability Analysis for Row-Based FPGAs”, *Proc. of ICCAD*, pp.230-235, 1993.
- [7] K. Zhu and D.F. Wong, “On Channel Segmentation Design for Row-Based FPGAs”, *Proc. of ICCAD*, pp.26-29, 1992.
- [8] S. Sastry and A.C. Parker, “Stochastic Models for Wireability Analysis of Gate Arrays”, *IEEE Trans. CAD*, vol.5, no.1, pp.52-65, January 1986.