行政院國家科學委員會專題研究計畫　成果報告

高效能之現場可規劃連接晶片之設計

計畫類別：個別型計畫
計畫編號：NSC91-2215-E-168-002-
執行期間：91 年 08 月 01 日至 92 年 07 月 31 日
執行單位：崑山科技大學電機工程系

計畫主持人：王炳聰

計畫參與人員：佘卓霖 、田朝元

報告類型：精簡報告

處理方式：本計畫可公開查詢

中　華　民　國 92 年 10 月 17 日

行政院國家科學委員會補助專題研究計畫 ■成果報告 □期中進度報告

高效能現場可規劃連接晶片之設計

A High Performance Field Programmable Interconnect Chip

計畫類別：■ 個別型計畫　□ 整合型計畫
計畫編號：NSC91 － 2215 － E － 168 － 002 －
執行期間： 91 年 8 月 1 日至 92 年 7 月 31 日

計畫主持人：王炳聰
共同主持人：
計畫參與人員： 佘卓霖、田朝元

成果報告類型(依經費核定清單規定繳交)：■精簡報告 □完整報告

本成果報告包括以下應繳交之附件：
□赴國外出差或研習心得報告一份
□赴大陸地區出差或研習心得報告一份
□出席國際學術會議心得報告及發表之論文各一份
□國際合作研究計畫國外研究報告書一份

處理方式：除產學合作研究計畫、提升產業技術及人才培育研究計畫、列管計畫及下列情形者外，得立即公開查詢
□涉及專利或其他智慧財產權，□一年□二年後可公開

查詢

執行單位：崑山科技大學電機系

中　華　民　國　92　年　10　月　1　日

中文摘要

關鍵詞：現場可規劃連接晶片、連接閂、配置、細部繞線

現場可規劃連接晶片(Field Programmable Interconnection chips; FPICs)所需之開關數量過多為此種晶片的最大缺點，爲克服此一缺點，本計畫提出一種高效能連接架構之 FPIC。此種連接架構是將一個大連接閂(Crossbar)分開為幾個可達到百分之百繞線率之較小連接閂，以減少晶片所需之開關數目，接著以層次性連接方式來連接這些較小之連接閂或所需連接之積體電路(ICs)，以構成一個完整之現場可規劃連接晶片。爲得到具高效能與高密度之連接架構，在相同繞線率之條件下，本計畫以量化之方式，來討論應將一個大的連接閂分成幾個較小的連接閂，以得到較具一般性之最佳結果。本計畫採用一些現有演算法來發展一套電腦輔助設計軟體，即採用 min-cut 之演算法，將彼此有較多連接之連接閂或 IC，擺置於較接近之位置，以完成配置(Placement)之工作，接著將配置後所得之結果，發展一套繞線(Routing)演算法，以完成此新連接架構之細部繞線(Detailed Routing)。最後我們採用一些工業上常用之電路來進行試驗，證明此架構可達百分之百之繞線率，以進一步強化此高效能現場可規劃連接晶片之實用性。

Abstract

Keywords：Field Programmable Interconnection Chip、Crossbar、Placement、Detailed Routing

Field programmable interconnection chips (FPICs) has very large number of programmable switches that is it drawback. In this project, a hierarchical crossbar interconnection structure for field programmable interconnection chip is proposed to overcome these shortcomings. They can reduce the total number of programmable switches in an FPIC which is divided a large full crossbar into a set of partial crossbars. The new interconnection structure is created with hierarchical partial crossbar interconnection, and each level consists of a set of partial crossbar to connect logic ICs or crossbars. To obtain the optimal structures with high performance and high density, various structures with the same routability are discussed. The field programmable interconnection chip with the new architecture can be efficiently configured with existing computer aided design algorithm. The *min-cut* algorithm is applicable to the placement. The routing paths are accomplished by picking a set of number for each cut net. Experiments on benchmark circuits show that the density is significantly improved and the 100% routability can be achieved.

1. Introduction

Field programmable interconnect chips (FPICs) have played a key role in the customized interconnection to take a large gate-level logic design and map it into hardware from by configuring a set of tens or hundreds of ICs [1-7]. The separation of the logic and interconnect can simplify the problem of placement and routing and can improve the system performance as a whole [2]. FPICs can facilitate hardware emulation, circuitry prototyping telephone communication, … etc. Since a high density FPIC can eliminate the need for

designing PCB for board-level designs, components on a board can be interconnected by configuring the FPIC.

The FPICs must provide reconfigurability and requires 100% routability, the interconnection structure differs from the conventional switch networks. Working in the interconnection structure of an FPIC in particular has focused on topics such as the total number of programmable switches and the number of programmable switches along signal paths under the 100% routibility.

Since the crossbar structure has the highest regularity and easy to design, this structure is commonly used in two commercially available FPICs, Aptix [6] and I-Cube [7]. The Aptix FPIC provides high degree of integration but the total number of programmable switches is more complexity. It architecture is similar to a symmetric FPGA with logic blocks substituted by the I/O pads. Each I/O pad is arranged in a L-shaped pattern and is attached by two internal I/O track. The programmable switch exists at the intersection of the two orthogonal wires. As for the I-Cube FPIC, it is very concise and makes use of minimal programmable switches. In the I-Cube, a connection between two I/O pins can be established by closing the switch at the intersection of the corresponding orthogonal signal lines. Every connection can be made or broken without affecting another. Therefore the I-Cube FPIC can provides 100% routability and predictable propagation delay. However, the total number of programmable switches used in these two FPICs are higher complexity.

In this paper, we will present a new interconnection structure FPIC. The total number of programmable switches in the new architecture is significantly reduced at a sacrifice of some performance. Various structures with the same routability are compared to obtain a optimal interconnection structure for FPICs with high density. The new architecture can reduce the placement and routing complexity, achieves bounded interconnection delay, scales

linearly with the pin count, and allows expansion to systems with hundreds of thousands of FPGA devices in a fast and uniform way.

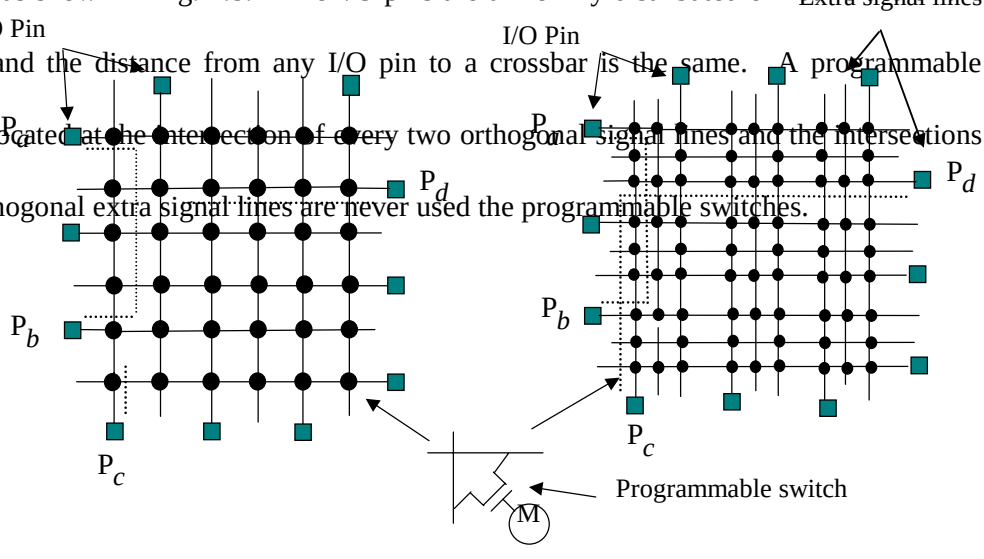
The field programmable interconnect chip with new partial crossbar interconnection structure may be efficiently configured with the existing CAD (computer aided design) algorithm. The placement can be used *min-cut* technique to minimize the number of cut nets that result from the partitioning. The routing paths are accomplished by picking a set number for each cut net.

This projectt is organized as follow: In section 2, we describe the modified mesh crossbar structure. The partial interconnection structure for FPICs is presented in Section 3. Section 4 presents conclusions.

2. The Crossbar Architecture

2.1 The Mesh Crossbar Structure

Fig. 1.a is the general crossbar structure. The structure has the highest regularity, but it cannot achieve 100% routability. For example, if a connection path for pin P_a and pin P_b is connected via the path for P_c will be occupied and cannot used for other routing requirement. The modified mesh crossbar structure proposed in [14] can achieve 100% routability by addition some extra signal lines between every two columns and rows lines as shown in Fig. 1.b. For example, a connection path for pin P_a and pin P_b uses an extra signal line, and an another connection path for pin P_c and pin P_d also has a path to be connected as shown in Fig. 1.b. The I/O pins are uniformly distributed on four sides of the crossbar, and the distance from any I/O pin to a crossbar is the same. A programmable switch is located at the intersection of every two orthogonal signal lines and the intersections of two orthogonal extra signal lines are never used the programmable switches.



(a) (b)

Fig. 1 a) The mesh crossbar cluster and b)The modified mesh crossbar cluster.

Let N_p and N_s be the number of I/O pins and number of programmable switches in a crossbar structure, respectively. Assume that I/O pins are uniformly distributed on four sides of a crossbar structure, thus the number of I/O pins in each side are equal to $N_p/4$. For the mesh cross structure, the relation between N_p and N_s will be

$$N_s = \left(\frac{N_p}{4} \cdot 2\right) \cdot \left(\frac{N_p}{4} \cdot 2\right) = \frac{{N_p}^2}{4}$$

As for the modified crossbar structure, the relation between N_p and N_s will be

$$N_s = \frac{N_p}{4} \cdot \frac{N_p}{4} \cdot 8 = \frac{{N_p}^2}{2}$$

2.2 A Full Crossbar structure Versus a Set of Partial Crossbars structure

All the I/O pins in an FPIC connected to only one crossbar structure is called a full crossbar structure. A full crossbar structure can provide the best performance of any interconnection system but the expense of complexity, chip size, and cost. Since the number of programmable switches used are proportional to the square the number of I/O pins and so is not economical. Although, a full crossbar structure can provides ideally reliable in performance, but the total number of programmable switches grows as the square of its pin counts. Since a programmable switches possesses higher resistance and higher capacitance

than regular metal wire and its use extra area. In this paper, a full crossbar structure is divided into a set of small crossbar clusters and using a new interconnection structure connect the small crossbar clusters. Thus the total number of programmable switches in the new FPIC architecture can be significantly reduced.

Comparing a full crossbar structure to construct an FPIC, the number of programmable switches used in the new structure is significantly reduced at sacrifice of some performance. For example, a full crossbar structure as shown in Fig. 1.b, is connected *N* I/O pins to construct an FPIC, the total number of programmable switches are equal to $\frac{N^2}{2}$. If we divide a full crossbar structure into *k* crossbar clusters to construct an FPIC with the new interconnection structure, the number of programmable switches are equal to $\frac{N^2}{2 \cdot k}$. Obviously, the total number of programmable switches in the new interconnection structure is inverse ratio of *k*. What FPIC interconnection structure to interconnect the set of small crossbar clusters can provide a scaleable architecture that can be expanded to fit design of all sizes. Therefore, a good interconnection structure is needed to accord with the requirements.

3. The New Structure FPICs

This section presents a new structure of an field programmable interconnect chip. Several interconnection structures to implement the new interconnection structure are defined. The total number of programmable switches is used to compare FPICs densities.

3.1 The New Partial Crossbar Interconnection Structure

A small crossbar clusters interconnection structure for FPIC had presents a in [2]. The small crossbar clusters interconnection structure consists of a set of small full crossbar, connected to logic FPGA's but not to each other. The I/O pins of each FPGA are divided into proper subset, using the same division on each one. The I/O pins of each crossbar chip are connected to the same subset of pins from each logic chip. Since the partial crossbars is

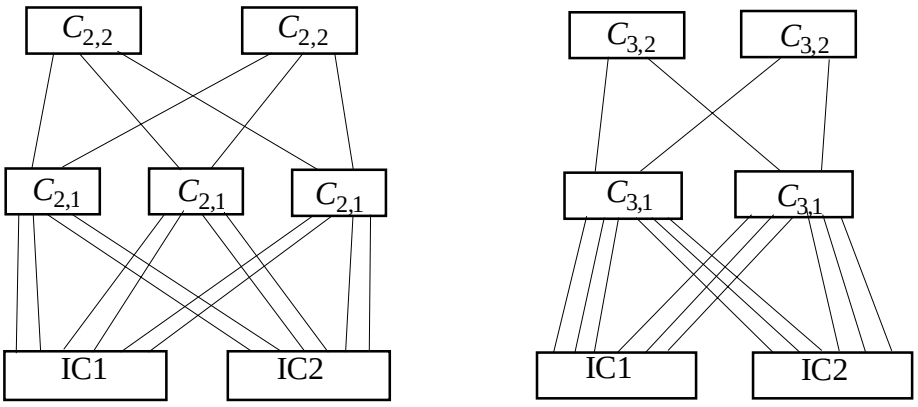
applied recursively to interconnect large system, the hierarchical levels of partial crossbar may be large. Thus the performance of the FPIC is significantly reduced.

In this paper, a new interconnection structure is created by two hierarchical levels partial crossbar interconnections, and each level consists of a set of partial crossbars to connect either logic FPGA's or partial crossbars but no to each other. Since a signal passing through a partial crossbar is fewer than 2 programmable switches, any two I/O pins in an HFPIC can be connected with fewer 6 programmable switches. The I/O pins of each partial crossbar are connected to the same subset of pins from each logic FPGA. Each subset is contained many pins, called a path. Fig. 3 illustrates two FPIC's with the new partial crossbar interconnection structures. A subset contains *two pins* and *three pins* is shown in Fig. 3.a and Fig. 3.b, respectively. An FPIC with the new partial crossbar interconnection structure is called *HFPIC*. It is denoted *q-HFPIC* if the I/O pins in each subset has q I/O pins.

Let m be the total number of I/O pins in an HFPIC. Also let $N_{c,1}$ be the number of crossbar clusters in the first hierarchical level. Assume that the number of I/O pins in each IC are the same. We can calculate $N_{c,1}$ as follows:

$$N_{c,1} = \frac{m}{q}$$

As q increases, larger I/O pins distribute in a partial crossbar and fewer partial crossbars needed. Since the programmable switches in a crossbar grows as the square of its pin counts, the total number of programmable switches in an HFPIC is related to q. Also we can set the number of partial crossbars in the second level to be equal to the number of FPGAs to construct the two hierarchical interconnection structure.



(a) q=2 (b) q=3

Fig. 3 The FPIC's with hierarchical small crossbar clusters interconnection structure.

The number of tracks in a partial crossbar is related to q and the number of programmable switches in a partial crossbar is related to the number of I/O pins distributed in a partial crossbar. Let $C_{q,i}$ be a partial crossbar in the ith hierarchical level associated with the q-HFPIC. An I/O pin of $C_{q,1}$ is either connected to logic FPGAs, called *child pins*, or to partial crossbars $C_{q,2}$, called *parent pins*. The *pin-connection ratio* of a small crossbar clusters $C_{q,1}$, denoted α_q, is defined to be the ratio of the number of parent pins to the number of child pins. Let m be the number of I/O pins in a IC. The child pins in a $C_{q,1}$ are equal to n subsets, that is, a partial crossbar $C_{q,1}$ has $q \cdot n$ child pins and $\alpha_q \cdot q \cdot n$ parent pins. Since structure of HFPIC is a two-level hierarchical interconnection, a partial crossbar $C_{q,2}$ has $\alpha_q \cdot m$ child pins.

To compare the HFPICs with each other, we must assume the various HFPICs have the same routability. The large is q, the more nets can be connected through the same crossbar cluster, hence α_q is inverse ratio of q. Let $\alpha_{q'}$ be the *pin-connection ratio* of a partial crossbar $C_{q',1}$ with a q'-HFPIC. As q increases, the more number of I/O pins can connect in a small crossbar structure. For the q'-HFPIC and the q-HFPIC to have the same routability,

$$\alpha_q \cdot q \cdot n = \alpha_{q'}^{\log_{q'} q} \cdot q \cdot n .$$

Therefor, we have

$$\alpha_q = \alpha_{q'}^{\log_{q'} q} .$$

For example, a *2*-HFPIC and a *3*-HFPIC are shown in Fig. 3.a and Fig. 3.b, respectively. For the same routability, we have

$$\alpha_3 \cdot q \cdot n = \alpha_2^{\log_2 3} \cdot q \cdot n .$$

Let α_2 be the *pin-connection ratio* of parent pins to child pins in a small crossbar cluster $C_{2,1}$ associated with *2*-HFPIC. Then,

$$\alpha_q = \alpha_2^{\log_2 q}.$$

Let $\beta = \log_2 q$. Then,

$$\alpha_q = \alpha_2^{\beta}.$$

For example, if a *3*-HFPIC has the same routability as a *2*-HFPIC with $\alpha_2 = 1$, the *pin-connection ratio* of a partial crossbar $C_{3,1}$ with a *3*-HFPIC, α_3 must be 1. Obviously, $0 < \alpha_2 \le 1$. If $\alpha_2 = 1$, then the total number of child pins and parent pins are the same in the partial crossbars $C_{2,1}$, that is, every child pins can respectively be connected to a parent pins. Equivalently, if $\alpha_2 = 1$, then $\alpha_q = \alpha_2^{\log_2 q} = 1$ for every partial crossbar $C_{q,1}$ in a *q*-HFPIC. If $\alpha_2 = 0.5$, the total number of parent pins in a partial crossbar $C_{2,1}$ in a 2-HFPIC is half of the total number of child pins, that is, in average one of *q* child pins in a partial crossbar $C_{q,1}$ in a *q*-HFPIC can be connected to a parent pin.

3.2 The Total Number of tracks in a *q*-HFPIC

Let N_{tq} be the total number of tracks in a *q*-HFPIC. We can calculate N_{tq} as follows:

$$N_{tq} = m \cdot n + \alpha_q \cdot m \cdot n, \text{ for } \alpha_q < 1, \text{ and}$$

$$N_{tq} = 2 \cdot m \cdot n, \text{ for } \alpha_q = 1.$$

By substitution,

$$N_{tq} = m \cdot n + \alpha_2^{\log_2 q} \cdot m \cdot n, \text{ for } \alpha_2 < 1, \text{ and}$$

$$N_{tq} = 2 \cdot m \cdot n, \text{ for } \alpha_2 = 1.$$

Fig. 4 illustrates the total number of tracks in a *q*-HFPIC versus q for $\alpha_2 = 1$, $\alpha_2 = 0.8$, $\alpha_2 = 0.5$, and $\alpha_2 = 0.3$ in an HFPIC with *m*=128 and *n*=10. As q increases, fewer tracks are needed to achieve the same routability. For the $\alpha_2 = 1$, the N_{tq} for the various q in an HFPIC is the same value. However, a track does not take large space than a programmable switch and extra logic circuits are needed to control a programmable switch. The complexity of the HFPIC chip is dominated by the total number of programmable switches. This will be discussed in the follow section.

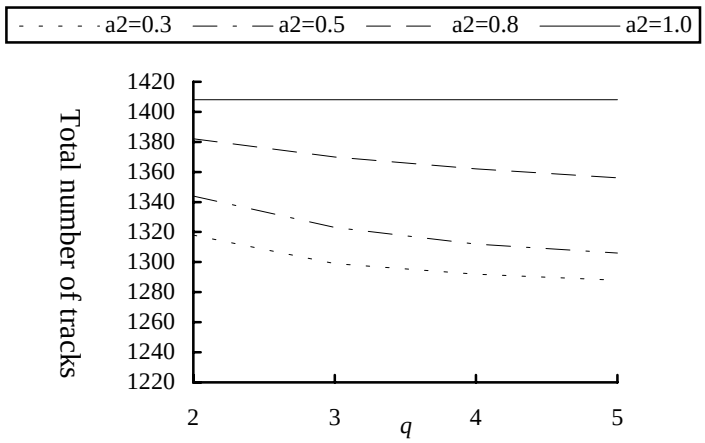


Fig. 4 The total number of tracks versus q for $\alpha_2 = 0.3$, $\alpha_2 = 0.5$, $\alpha_2 = 0.8$, and $\alpha_2 = 1.0$ in an HFPIC with *m*=128 and *n*=10.

3.3 The Total Number of Programmable Switches in a *q*-HFPIC

The subsection is calculated the total number of programmable switches in a *q*-HFPIC. Let N_{ST} be the total number of programmable switches in a *q*-HFPIC. Recall that α_q is the *pin-connection ratio* of parent pins to child pins in a partial crossbar $C_{q,1}$, m is the number of I/O pins in a logic FPGA, and an HFPIC is connected n FPGAs. We can calculate N_{ST} as follows:

$$\#\text{ of } C_{2,1} \text{ is equal to } \frac{m}{q}$$

$$\#\text{ of I/O pins in } C_{q,1} \text{ is equal to } q \cdot n + \alpha_q \cdot q \cdot n$$

$$\#\text{ of I/O pins in } C_{q,2} \text{ is equal to } \alpha_q \cdot q \cdot n$$

$$N_{ST} = \left[\frac{\left(q \cdot n + \alpha_q \cdot q \cdot n\right)^2}{2} \right] \cdot \frac{m}{q} + \frac{\left(\alpha_q \cdot q \cdot n\right)^2}{2} \cdot n$$

$$= \frac{q \cdot n^2 \left[m\left(1+\alpha_q\right)^2 + \alpha_q^2 \cdot q \cdot n \right]}{2}$$

$$N_{ST} = \frac{q \cdot n^2 \left[m \cdot \left(1+\alpha_q\right)^2 + {\alpha_q}^2 \cdot q \cdot n \right]}{2} \quad \text{for } m >> q.$$

By substitution,

$$N_{ST} = \frac{q \cdot n^2 \left[m \cdot \left(1+\alpha_2^{\beta}\right)^2 + \alpha_2^{\beta^2} \cdot q \cdot n \right]}{2} \quad \text{for } \alpha_2 < 1, \text{ and}$$

$$N_{ST} = \frac{q \cdot n^2 \cdot \left(4m + q \cdot n\right)}{2} \quad \text{for } \alpha_2 = 1.$$

The total number of programmable switches in a HFPIC can be used to compare the various HFPIC's with the same routability. Table 1 shows four HFPIC's with different interconnection structures, q=2, 3, 4, 5. Assume all four HFPIC's have the same routability, each HFPIC is connected *10* logic FPGAs and each FPGA has *128* I/O pins. Each column of Table 1 shows the value of a parameter, and each row is associated with a HFPIC. The last column of Table 1 shows the total number of programmable switches in each HFPIC.

In Table 1, *3*-HFPIC needs the minimum number of programmable switches for the all *q*-HFPICs, and it is needs 43236 programmable switches. If a large full crossbar is used in this case, it is needs 819200 programmable switches. Take the worst case, $\alpha_3 = 1$, for *3*-HFPIC, the structure needs 158720 programmable switches. The ratio of total number of programmable switches used for the new architecture with the worst case and a large full crossbar is about one-fifth. Hence, the total number of programmable switches in the new architecture is significantly reduced.

Table 1 The total number of programmable switches for the various HFPIC's with the same routability.

	α_q	N_{ST}
q=2	0.5	49280
q=3	0.33	43236
q=4	0.25	45120
q=5	0.2	49357

Since the total number of programmable switches is function of q. The selection of the value for the variable depend on the characteristics of the circuits to be implemented and the desired performance of circuits. Fig. 5 illustrates the total number of programmable switches versus α_2 for various HFPIC's.

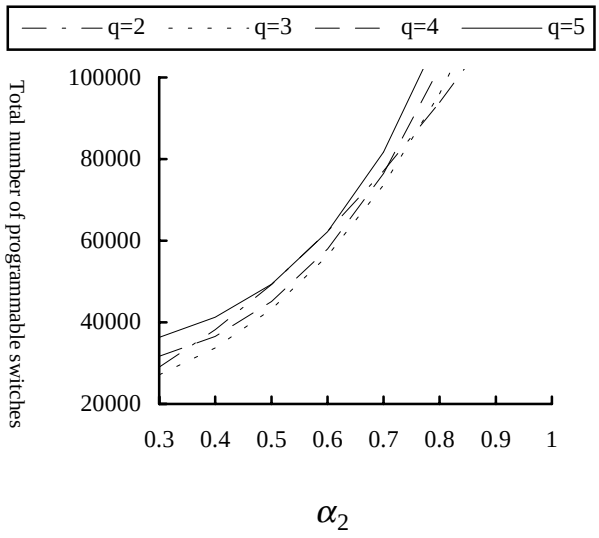


Fig. 5 The total number of programmable switches in the HFPIC's with n=10, m=128 versus α_2.

Since the HFPIC is a two hierarchical level interconnection structure, the performance of the various HFPIC's are equality. The problem of obtaining the optimal structure of HFPIC is only determined by q. Recall that as q increases, larger I/O pins distribute in a partial crossbar and fewer partial crossbar needed for the q-HFPICs. Since the programmable switches in a crossbar grows as the square of its pin counts, the total number of programmable switches is related to q. The value of α_2 depends on the characteristics of circuits and the placement of FPGA's. Investigation on α_2 was performed experimentally

by implementing a set of industrial circuits as HFPIC's. An α_2 in the range between 0.4 and 0.6 is preferred.

Since the performance in the *q*-HFPIC's is equality, the optimal structure are only considering the density of an HFPIC. By considering the routing completeness of partial circuits, *q*=3 is the most appropriate choice in the density for the HFPIC's.

4. Conclusions

A new partial crossbar interconnection structure for field programmable interconnect chip is presented. A large full crossbar divide into a set of partial crossbars to construct the two level hierarchical interconnection structure for an FPIC. The performance of the various FPGAs structure is equality. To obtain the optimal structure of FPIC's depend on the densities. The total number of programmable switches in an HFPIC is significantly reduced at a sacrificed of some performance. Various structure for HFPICs with equivalent routability are discussed to obtain a optimal interconnection structures. Empirical results of benchmark circuits show HFPIC's require fewer programmable switches than other FPIC structures.

References

[1] Bhatia. D, Haralambides. J, "Resource requirements and layouts for field programmable interconnection chips," IEEE Trans., on VLSI, vol 3, no. 3, Jue 2000, pp345-355.

[2] Mao-Hsu Yen, Sao-Jie Chen, Lan, S.H., "Symmetric and programmable multi-chip module for rapid prototyping system," IEEE Workshop on Signal Processing Systems, 1999, pp301-310.

[3] Mao-Hsu Yen, Mon-Chau Shie, Lan, S.H., "Polygonal routing network for FPGA/FPIC," VLSI Technology, Systems, and Applications, 1999, pp104-107.

[4] Wen-Jong Fang, Wu A.C.-H., "A hierarchical functional structuring and partitioning approach for multiple-FPGA implementations," IEEE Trans., on CAD, vol 16, no. 10,

Oct. 1997, pp1188 -1195.

[5] M. Butts, J. Batcheller and J. Varghese, “An efficient logic emulation system,” *Proc., ICCD*, pp. 138-141, 1992.

[6] T. M. Butts, J. Batcheller and J. Varghese, “An efficient logic emulation system,” *IEEE Trans., on VLSI*, Vol. 1, No. 2, pp. 171-174, June 1993.

[7] P. K. Chan and Martine D. F. Schlag, “Architecture tradeoffs in filed-programmable device-based computing systems,” *IEEE Workshop on FPGA for Custom Computing Machine*, pp. 152-161, 1993.

[8] D. Bhatia and J. haralambides, “Resources requirements for filed programmable interconnection chips,” *Proc., IEEE 8th International Conference VLSI Design*, pp. 376-380, 1995.

[9] S. Hauck, G. Borriello, and C. Ebeling, “Mesh routing topologies for multi-FPGA system,” *Proc., ICCD*, pp. 170-177, 1994.

[10] The I-Cube FPID family data books, I-Cube, Inc., Nov. 1993.

[11] C. Kim and H. Shin, “A performance-driven logic emulation system: FPGA network design and performance-driven partitioning,” *IEEE Trans., on CAD*, Vol. 15, No. 5, pp. 560-568, May 1996.

[12] K. Zhu, D.F. Wong, and Y. W. Chang, “Switch module design with application to two-dimensional segmentation design,” *Proc., ICCAD*, pp. 480-485, 1993.

[13] Y. W. Chang, D. F. Wong, and C. K. Wong, “Design and analysis of FPGA/FPIC switch modules,” *Proc., ICCD*, pp.394-401, 1995.

[14] P. T. Wang, K. N. Chen, and Y. T. Lai, "A high performance FPGA with hierarchical interconnection structure," *Proc. ISCAS'94*, pp. 4.239-4.242.

[15] Y. T. Lai and P. T. Wang, "Hierarchical interconnection structure for field programmable gate arrays," *IEEE Trans. on VLSI*, vol. 5, no. 2, June 1997, pp. 186-196.

[16] A. A. Aggarwal and D. M. Lewis, “Routing architectures for hierarchical filed programmable gate arrays,” *Proc., ICCD*, pp. 475-478, 1994.

[17] S. F. Juang, “A field programmable interconnect chip,” Master thesis, Department of Information Engineering, National Tsing Hua University, 1995.

[18] Y. Sun, T. C. Wang, C. K. Wong, and C. L. Liu, “Routing for symmetric FPGA’s and FPIC’s,” IEEE *Trans. on CAD* vol. 16, no. 1, Jan. 1997, pp. 20-31.