

# Fast Flip-Chip Pin-Out Designation Respin for Package-Board Codesign

Ren-Jie Lee, *Student Member, IEEE*, and Hung-Ming Chen, *Member, IEEE*

**Abstract**—Deep submicrometer effects drive the complication in designing chips, as well as in package designs and communications between package and board. As a result, the iterative interface design has been a time-consuming process. This paper proposes a novel and efficient approach to designating pin-out, which is a package ball chart describing pin locations for flip-chip BGA package when designing chipsets. The proposed approach can not only automate the assignment of more than 200 input/output (I/O) pins on package, but also precisely evaluate package size which accommodates all pins with almost no void pin positions, as good as the one from manual design. Furthermore, the practical experience and techniques in designing such interface has been accounted for, including signal integrity, power delivery and routability. This efficient pin-out designation and package size estimation by pin-block design and floorplanning provides much faster turn around time, thus enormous improvement in meeting design schedule. Our pin-block design contains two major parts. First, we have pin-block construction to locate signal pins within a block along the specific patterns. Six pin patterns are proposed as templates which are automatically generated according to the user-defined constraints. Second, we have pin-blocks grouping to group all pin-blocks into package boundaries. Two alternative pin-blocks grouping strategies are provided for various applications such as chipset and field-programmable gate array (FPGA). The results on two real cases show that our methodology is effective in achieving almost the same dimensions in package size, compared with manual design in weeks, while simultaneously considering critical issues and package size migration in package-board codesign.

**Index Terms**—Package-board codesign, pin-block floorplanning, pin-out designation.

## I. INTRODUCTION

**B**ECAUSE of deep submicrometer (DSM) technology, chips now contain more functionality and are being driven to higher performance levels than ever before. Consequently, with more functionality on the chip, designers have to deal with higher input/output (I/O) densities, more signals coming out of a chip and tighter geometries [9]. This leads to the complication in designing package which accommodates chips, as well as the board which accommodates the packages. As a result,

the ability to design the chip, the package, and surrounding system concurrently becomes a primary advantage, but also a challenge.

Recently chip-package codesign has drawn attention under these circumstances, for example [3], [12]. However package-board codesign, which is definitely not a trivial work, still needs more works. Several works [5], [13], [4], [2] which were related to package and printed circuit board (PCB) physical designs. Reference [5] presented a style for ball grid array (BGA) ball-out, but shielding pins used for preventing pin-to-pin crosstalk were not considered. Moreover, when they try to keep the package cost small, this style will put a restriction on the maximum package size. Thus, there is a limit to the number of BGA balls that can be used for power delivery, and area for power delivery from motherboard to package. Reference [13] proposed an algorithm which assigned and routed the solder bumps of a BGA package to a set of fan-out points in a single layer. This work only created a topological routing, not precise geometry layout, and only the routability issue on PCB is considered.

For pin assignment problem, [4] presented a simulated annealing algorithm to find a pin assignment solution which considered the routability issue on BGA package and PCB, but no other DSM effects were considered. Reference [2] suggested a direction of research for topological pin assignment. The two-stages heuristic algorithms, initial pin assignment and assignment improvement, can be closely attuned to a specific router, then enhance the routability of PCB by reducing wiring congestion and path crossovers. Since this methodology disregarded the package pin number, it can not be applied to assign the pin-out of flip-chip BGA package which has large number of pins and significant cost issue. All these researches are not suitable for modern package-board codesign, which is requested to have minimal turn around time and optimized signal performance as well as package cost.

Fig. 1(a) shows the typical interface design flow for IC-package-PCB codesign. In general, IC designers finish the pin designation based on experience (rule-of-thumb). In order to tradeoff signal performance and package cost, they always take few weeks to modify package size, rework package substrate and PCB layout, then rearrange pin-out. This conventional process can not efficiently estimate an accurate package size during designating pins for flip-chip BGA and possibly degrade signal performance due to the weakness on product experience and basic design concept. Furthermore, these costly rework constantly postpone the schedule of chip implementation, thus lengthen the time to market (TTM). Fig. 1(b) illustrates our proposed design flow. To begin with, we will

Manuscript received May 02, 2007; revised October 26, 2007 and March 14, 2008. First published June 02, 2009; current version published July 22, 2009. This work was supported in part by the National Science Council of Taiwan ROC under Grant NSC 97-2220-E-009-001 and Grant NSC 97-2221-E-009-174-MY3.

The authors are with the Department of Electronics Engineering, National Chiao Tung University, Hsinchu 300, Taiwan (e-mail: rjlee@vda.ee.nctu.edu.tw; hmchen@mail.nctu.edu.tw).

Digital Object Identifier 10.1109/TVLSI.2009.2017795

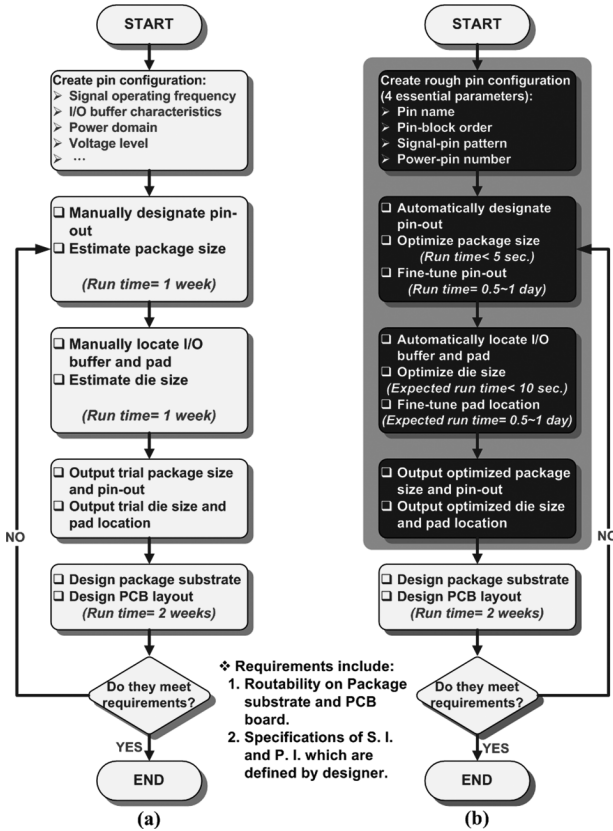


Fig. 1. (a) Typical flow and (b) proposed approach in interface design planning for IC-package-board codesign. Our approach will significantly shorten the runtime throughout the automation process. It cannot only automate pin-out designation efficiently, but also optimize package size during design stage, thus reduce the time of iteration.

create the rough pin configuration, which includes only four essential parameters. We automatically determine the pin-block order through an intuitive manner and flexibly design an appropriate pin pattern by solving ILP problems. After finishing the pin configuration, the designers will obtain the optimized package size and die size by automatically designating pin-out and locating I/O buffers. This paper aims at presenting a novel approach to designating pin-out and replacing heavy-loaded human design by automation process, which accounts for practical experience and techniques.

In this paper, we have formulated feasible constraints for automatically designing pin patterns, which are used to assign the signal pins along the particular constraints and work as templates. And then we proposed six signal-pin patterns for pin-block construction in package design. Signal integrity, power delivery, and routability have been accounted for in those patterns. This helps to speed up the process of pin-out designation. Furthermore, we have proposed a near optimal approach to minimizing package size by mathematical (linear) programming formulation. The package size migration issues are also considered through a simple estimation. The experimental results show that our solution can achieve almost the same results as manually designed by experienced designers, with much less time.

The remainder of the paper is organized as follows. Section II discusses the flip-chip BGA package design with PCB and

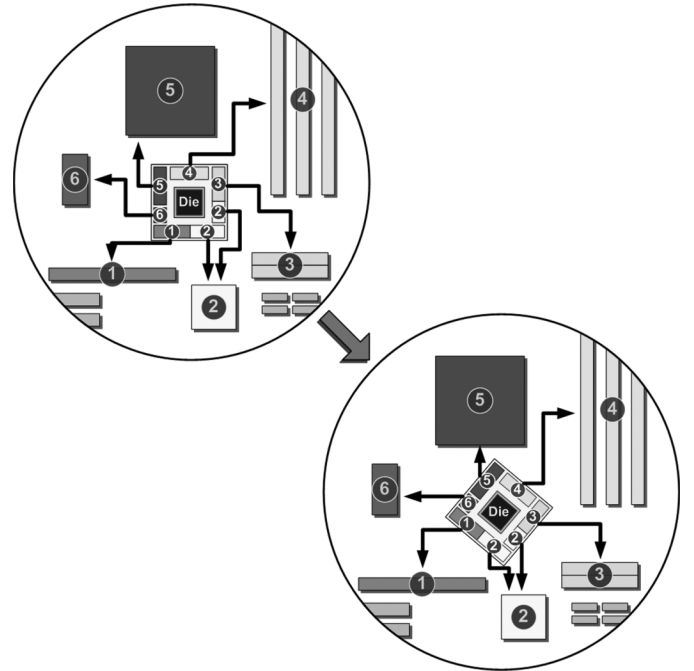


Fig. 2. General layout of PCB board. The order of pin-blocks on IC package should be assigned according to the corresponding components then fine-tuned the direction of package to meet minimum net-length.

DSM effects consideration, while Section III describes our pin-out designation by near optimal planning in package size. Section IV shows the experimental results, followed by the conclusion in Section V. An extended abstract of this paper with less detailed automation processes and without the alternative strategies for pin-blocks grouping and the considerations of package size migration issues (in Section III) was published by the ASP-DAC 2007 [8].

## II. PIN-OUT DESIGNATION BY CONSIDERING SIGNAL INTEGRITY AND POWER DELIVERY IN PACKAGE-BOARD CODESIGN

When we are designating pin-out, there are several critical constraints and considerations we need to take care of. In this section, we will discuss them comprehensively then introduce the design of pin pattern which can take all the constraints and considerations into account.

### A. Constraints and Considerations

1) *Locations of PCB Components:* Fig. 2 depicts a sketch of PCB layout. Usually PCB board contains several kinds of components and connectors which are applied to specific interfaces. The length of signal net from package pin to component or connector on PCB is the primary contributor to parasitic inductance. Therefore, package pins will exacerbate simultaneous switching noise (SSN) by increasing the parasitic inductance in the signal nets [7]. The familiar equation shown below describes the basic mechanism of SSN ( $V_{SSN}$ ):

$$V_{SSN} = NL_{tot} \left( \frac{dI}{dt} \right) \quad (1)$$

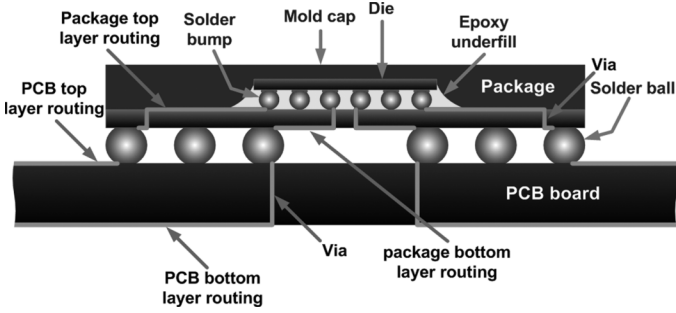


Fig. 3. Simplified cross-section of a flip-chip package which is mounted on PCB board.

where  $N$  is the number of switching drivers,  $L_{tot}$  is the equivalent inductance in which current must pass through, and  $I$  is the current per driver. In order to minimize the physical length of the signal net and thus reduce the total parasitic inductance, package pins should be accommodated in particular regions. As shown in Fig. 2, the minimum net-length can be obtained by assigning the order of pin-blocks according to the certain location of corresponding components or connectors then fine-tuning the direction of package properly.

2) *Routability*: Another crucial factor of successful pin designation is routability. For routing issue, the inflexible package-board routing rules force the row number of signal pins, signal net width and spacing on PCB to be critical constraints. Fig. 3 shows the simplified cross-section of a flip-chip package which is mounted on PCB board. For a general 4-layer PCB board, only the top and bottom layers are allowed to be routed nets; the second and third layers are used for planning power/ground plane. Based on the rules of thumb, package outer pins (solder balls located close to the package edge) connect solder bumps through vias and package top layer routing. These outer pins are then inevitably routed on PCB top layer. On the other hand, package inner pins located around the core of package must connect solder bumps by package bottom layer routing and then are routed on PCB bottom layer. Fig. 4 demonstrates the routing pattern on PCB top layer and package bottom layer respectively. For instance, when the diameter of PCB pad is 14 mil (1 mil = 25.4  $\mu\text{m}$ ), pad pitch is 39.37 mil, signal net width and spacing are both 5 mil on a 4-layer PCB board, the space between two pads can only be penetrated by two nets. It means only three rows of signal pins can be fanned out nets on PCB top layer. Because of these routing rules, the excess row number of signal pins will undoubtedly cause routing congestion due to restricted area between pins. Fig. 5 lists the confined row number of signal pins is constant and independent of the package sizes. In our example, the maximum row number of outer pins is nine and that of signal pins is seven (this happens when the nets on PCB bottom layer can be connected to those four rows of signal pins).

3) *Signal Integrity*: According to the routing pattern, shown in Fig. 4, we can generalize the rule of thumb in assigning pins. That is, if signal pins are allocated on the same row, their nets can have balanced routing, which means these nets will have matched impedance on PCB and package layout. On the other hand, if signal pins are allocated on the same column, there are only some nets which can have balanced routing. The matched

impedance is an essential requirement for high speed differential systems, because it can eliminate the common mode noise thus improve the signal performance. For signal integrity reason, return path inductance is another main course. The unsuitable placement and number of return path pins, which are power or ground pins, will maximize current return loops and increase return path inductance. This will dramatically degrade signal integrity and exacerbate radiated emissions. Its mechanism is similar to that of SSN and has been shown in (1). In regard to crosstalk noise, one of the major root causes is mutual capacitance [7], mainly because it will inject a current onto the neighbor victim pins. The induced noise ( $I_{\text{noise}, C_m}$ ) is proportional to the mutual capacitance ( $C_m$ ) and the rate in change of voltage on driven pins ( $dV_{\text{driver}}/dt$ )

$$I_{\text{noise}, C_m} = C_m \left( \frac{dV_{\text{driver}}}{dt} \right). \quad (2)$$

Therefore, the optimal pin designation is to place signal pin and power/ground pin proximally close to each other, so that each signal pin can be tightly coupled to a return path pin. This will minimize the effect of the return path inductance. Furthermore, if signal pins surrounded with ground pins, the mutual capacitance will be decreased and the noise is shielded extremely. In [6], [10], and [11], the effects of shielding, return path and reference plane are considered in package and PCB designs. However, those optimized designs, in terms of signal integrity concern, will create signal-pin blocks which have more power/ground pins but fewer signal pins within a large block area. The feasible designs of pin pattern are proposed in Section II-B.

## B. Pin Pattern Design

In order to automatically and flexibly design an appropriate pin pattern- we formulate the design constraints discussed in last section as feasible ILP problems. Therefore, we can obtain the proper pin patterns after solving the following ILP problems:

$$p_{j,k} = \begin{cases} 1, & \text{for signal pins} \\ 0, & \text{for power/ground pins} \end{cases}, \forall PA_i \quad (3)$$

$$\sum_{j=1}^{\text{row}} p_{j,k} \leq C_k, \forall PA_i, C_k \in \mathbb{N} \quad (4)$$

$$\sum_{k=1}^{\text{col}} p_{j,k} + p_{j,k+1} \leq D_j, \exists PA_i, D_j \in \mathbb{N} \quad (5)$$

$$\frac{SN_i}{\text{col} \cdot \text{row} - SN_i} \leq \text{SRR}_i, \forall PA_i \quad (6)$$

$$\frac{p_{j,k}}{4 - (p_{j+1,k} + p_{j-1,k} + p_{j,k+1} + p_{j,k-1})} > \leq \text{SSR}_i, \exists PA_i \quad (7)$$

$$\text{RPT}_i = \begin{cases} 1, & \text{for using power pins} \\ 0, & \text{for using ground pins} \end{cases}, \forall PA_i \quad (8)$$

where  $SN_i (= \sum_{k=1}^{\text{col}} \sum_{j=1}^{\text{row}} p_{j,k})$ , row and col are the signal-pin number per pattern, row number and column number of a pattern, respectively.

Equation (4) is the signal pin capacity ( $C_k$ ). It confines the signal pin number within a column for all patterns ( $PA_i$ ). As our previous discussion, the average number of this value is six. Equation (5) is the differential signaling constraints ( $D_j$ ).

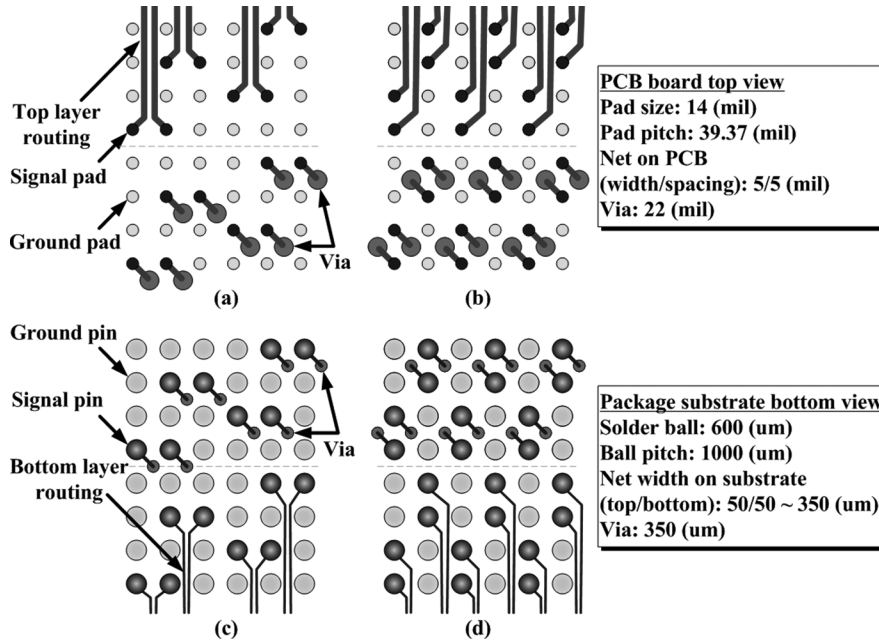


Fig. 4. (a), (b) Routing pattern on PCB top layer and (c), (d) package bottom layer. Because of the routing rules and restricted area between pins, the confined row number of signal pins is six. The excess row number of signal pins will cause routing congestion during the package substrate and PCB routing phase.

| Package size (mm) (Width x Height) | Pin number (Row x Column) | Row number of outer-pin (power-pin, ground-pin and signal-pin) |      | Row number of outer-pin (signal-pin only) |      |
|------------------------------------|---------------------------|--|------|---|------|
|                                    |                           | Max.   | Avg. | Max.                                      | Avg. |
| 37.5 x 37.5                        | 36 x 36                   | 9  | 8    | 7   | 6    |
| 35 x 35                            | 34 x 34                   | 9  | 8    | 7   | 6    |
| 31 x 31                            | 30 x 30                   | 9  | 8    | 7   | 6    |
| 27 x 27                            | 26 x 26                   | 9  | 8    | 7   | 6    |
| ...                                | ...                       | 9  | 8    | 7   | 6    |

Fig. 5. Restricted row number of signal-pin is constant and independent of package size due to inflexible package-board routing rules (PCB pad = 14 mil, pad pitch = 39.37 mil, net width = 5 mil, net spacing = 5 mil, for four layer PCB board).

The differential signal pins which exist in specific patterns must be strictly assigned at adjacent location in the same row (e.g.,  $p_{j,k+1} = 1$ , iff  $p_{j,k} = 1$ ). Equation (6) is the ratio of signal-to-return path pin ( $SRR_i$ ). The return path pins play an import role in signal integrity considerations, designers must define the essential ratio for each pattern according to its applications. Equation (7) is the ratio of signal-to-shielding pin ( $SSR_i$ ). For the purpose of isolating crosstalk noise, designers can set higher ratio of signal-to-shielding pin to assign ground pin in the neighboring location of signal pin. Otherwise, the ratio can be disregarded for low cost consideration. Obviously, these two ratios  $SRR_i$  and  $SSR_i$  will significantly trade off the performance and cost when we are designing pin patterns. Equation (8) is the type of return path pin ( $RPT_i$ ). Once the type of return path pin match that of PCB reference plane, the return path will induce the lower parasitic inductance [7]. Hence, this constraint should be defined along the type of reference plane (power/ground) on PCB.

For two layers PCB routing, Table I proposes six sets of constraints for generating six options of signal-pin patterns ( $PA_{i0}$  and  $PA_{i1}$  represent the fore-half and back-half of patterns). There exists tradeoff between signal performance and package

TABLE I  
CONSTRAINTS OF PROPOSED PIN PATTERNS

|                      |           | $C_k$ | $D_j$ | $SRR_i$ | $SSR_i$ | $RPT_i$ |
|----------------------|-----------|-------|-------|---------|---------|---------|
| Pattern 1 ( $PA_1$ ) | $PA_{10}$ | 6     | 2     | 1/2     | 1/3     | 0       |
|                      | $PA_{11}$ | 6     | 2     | 1/2     | 1/3     | 0       |
| Pattern 2 ( $PA_2$ ) | $PA_{20}$ | 6     | 2     | 1/2     | 1/3     | 0       |
|                      | $PA_{21}$ | 6     | N/A   | 1       | 1/3     | 0       |
| Pattern 3 ( $PA_3$ ) | $PA_{30}$ | 6     | N/A   | 1       | 1/3     | 0       |
|                      | $PA_{31}$ | 6     | 2     | 1/2     | 1/3     | 0       |
| Pattern 4 ( $PA_4$ ) | $PA_{40}$ | 6     | N/A   | 1       | 1/3     | 0       |
|                      | $PA_{41}$ | 6     | N/A   | 1       | 1/3     | 0       |
| Pattern 5 ( $PA_5$ ) | $PA_{50}$ | 6     | N/A   | 1       | 1/3     | 1       |
|                      | $PA_{51}$ | 6     | N/A   | 1       | 1/3     | 0       |
| Pattern 6 ( $PA_6$ ) | $PA_{60}$ | 6     | N/A   | 3       | N/A     | 1       |
|                      | $PA_{61}$ | 6     | N/A   | 3       | N/A     | 0       |

cost. Fig. 6 illustrates these proposed pin patterns and their simplified impedance models. The impedance of each net is composed of three components: serial resistor, serial inductor and shunt capacitor ( $Z_L = R + j\omega L + 1/j\omega C$ ). The first signal-pin pattern depicts that each pair of differential signal has been surrounded by ground pins. These ground pins can be performed as adjacent return path pins to minimize total inductance and

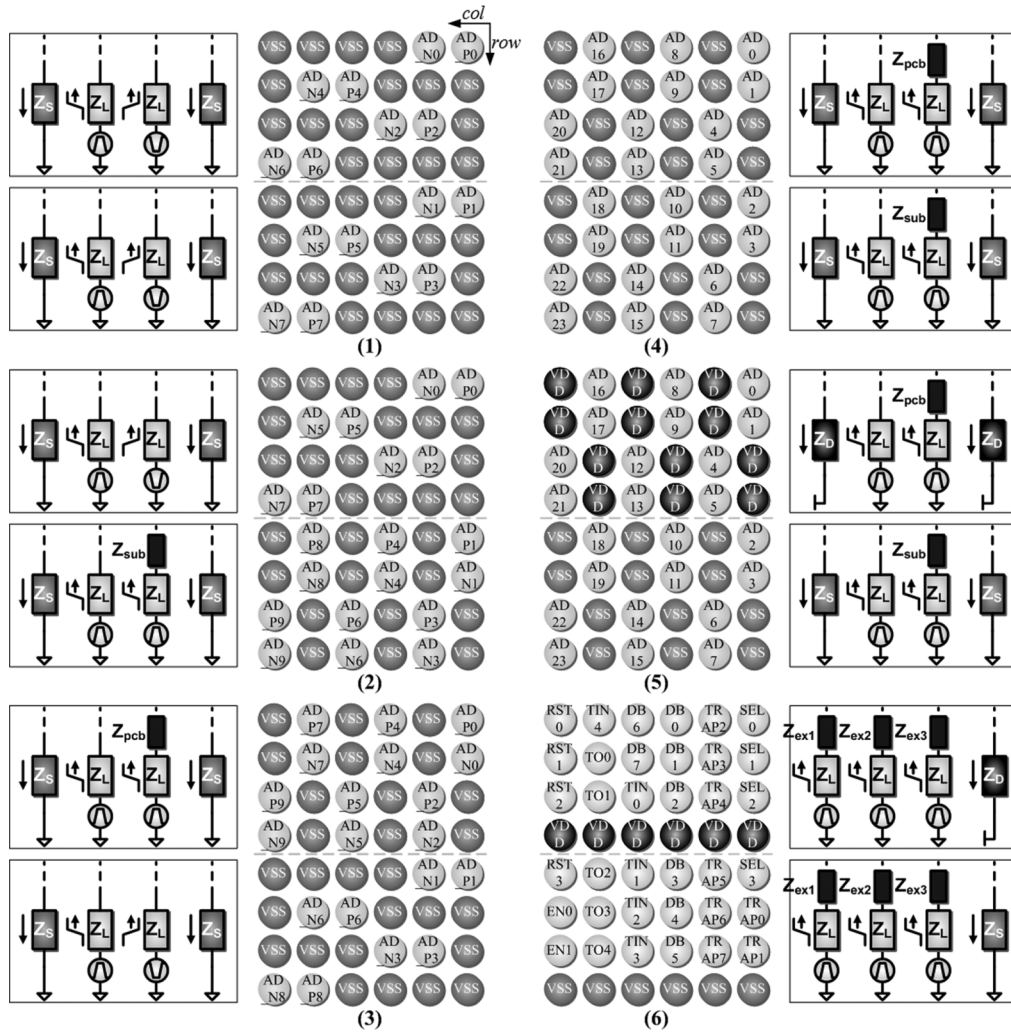


Fig. 6. Six pin patterns proposed in this paper. There exists tradeoff between routability and signal integrity concerns. The first pin pattern has better signal integrity, while the sixth one has the most efficient pin designation. In those patterns, AD\_P0/AD\_N0 is for differential signal (high speed), AD is for single-ended signal (high speed), and SEL or TRAP in sixth pattern are for single-ended signal (low speed or long-pulse signal).

as shielding pins to isolate pin-to-pin crosstalk noise. Moreover, the primary concern of differential system is on impedance-matching of nets. The first pattern has an exclusive advantage of nets balancing on PCB as well as package substrate layout, shown in Fig. 4(a) and (c). Thus, it is optimal for differential signals from the performance perspective, and can be modeled by two nets with matched impedance  $Z_L$ , as shown in Fig. 6. The only disadvantage of this pattern is poor pin designation efficiency.

In most cases, if the return current of a signal pin flows on ground planes, it should be coupled to ground pins to result in minimum return path, or vice versa. Whether a signal is coupled to just one power pin or just one ground pin, this case will emerge from the particular signal type and its configuration. Therefore, the fourth and fifth signal-pin patterns are proposed to provide two options for specific bus. The fifth pattern has better power delivery characteristic than the fourth one because of locating power pins. These two patterns arrange pins more efficiently than first pattern, but they both have worse signal integrity on PCB top-layer-routing and package bottom-layer-routing due to poor impedance-matching, shown

in Fig. 4(b) and (d). Hence, the net of each signal pair in its model has additional impedances except  $Z_L$  on PCB board ( $Z_{pcb}$ ) or on package substrate ( $Z_{sub}$ ). Both of them include extra equivalent resistance, inductance and capacitance. As compared with previously mentioned patterns, the second and third patterns are the compromises between signal performance and package cost. As for the sixth signal-pin pattern, it is the most efficient pin designation among all patterns because it contains the most signal pins than other patterns. The major disadvantage of this pattern is that it ignores all signal integrity concerns and can only be applied to test-in, test-out, or long-pulse control signal, which has less sensitivity in crosstalk. Therefore, its impedance model depicts these characteristics by using an undesirable and unpredictable impedance  $Z_{ext}$ , which is induced from PCB board and package substrate.

According to the experiences and basic concept of signal integrity, these six patterns have been characterized and shown in Fig. 7. Designers can take these patterns as templates and easily choose a specific pattern along the specification of individual bus, or they can design pin patterns which has sensible

|           | Application                               | Signal-pin NO. | Pin-to-pin crosstalk immunity | Net balance |              |                   |              | Signal shielding on package substrate (VDD/VSS) |              | Power delivery aware | Pin-designation efficiency |
|-----------|---|----------------|-------------------------------|-------------|--------------|-------------------|--------------|---|--------------|----------------------|----------------------------|
|           |   |                |                               | PCB board   |              | Package substrate |              | Top layer                                       | Bottom layer |                      |                            |
|           |   |                |                               | Top layer   | Bottom layer | Top layer         | Bottom layer |   |              |                      |                            |
| Pattern 1 | Differential signal                       | 16             | Excellent                     | Good        | Good         | Good              | Good         | VSS   | VSS          | Without              | Not good                   |
| Pattern 2 | Differential signal / Single-ended signal | 20             | Good                          | Good        | Good         | Good              | Not good     | VSS   | VSS          | Without              | Average                    |
| Pattern 3 | Differential signal / Single-ended signal | 20             | Good                          | Not good    | Good         | Good              | Good         | VSS   | VSS          | Without              | Average                    |
| Pattern 4 | Differential signal / Single-ended signal | 24             | Excellent                     | Not good    | Good         | Good              | Not good     | VSS   | VSS          | Without              | Good                       |
| Pattern 5 | Differential signal / Single-ended signal | 24             | Excellent                     | Not good    | Good         | Good              | Not good     | VDD   | VSS          | With                 | Good                       |
| Pattern 6 | Single-ended signal                       | 36             | Not good                      | Not good    | Not good     | Not good          | Not good     | None  | None         | With                 | Excellent                  |

Fig. 7. Characteristics of signal-pin patterns. According to the properties and requirements of specific signal, we can select a proper pattern to designate pins.

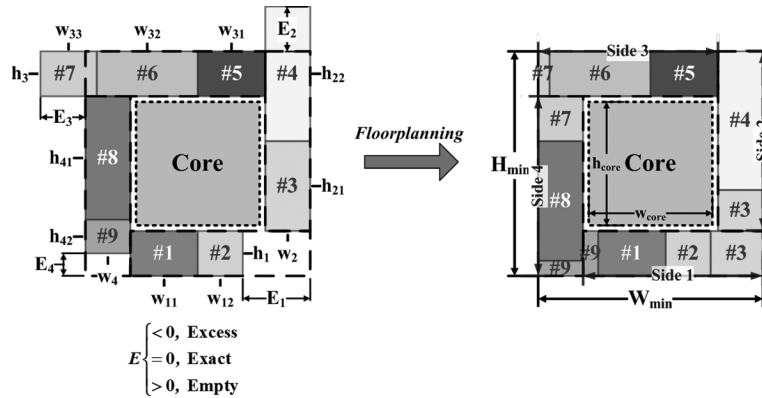


Fig. 8. Minimum package size can be obtained after we designate and floorplan all pin-blocks.

efficiency, routability, and signal integrity for their specific purposes by defining their dedicated constraints.

### III. FAST PIN-OUT DESIGNATION AUTOMATION BY PIN-BLOCK CONSTRUCTION AND FLOORPLANNING

By using those pin patterns, pin-blocks can be constructed and grouped for pin-out designation. In addition, package size will be minimized by pin-block floorplanning. This section presents the detailed strategies and methodologies. The package size migration issues will be considered in this section as well.

#### A. Pin-Block Construction and Grouping

In general, designers always take half or one day to define the complete pin configuration for high pin-count chip. It is because the most precise pin configuration will contribute the optimal pin-out and package size in manual design, but it is an exhaustive and time-consuming work. In our approach, the runtime of this manual job can be reduced by a rough pin configuration which simply contains four essential parameters: signal-pin name, pin-block placement sequence (order), selected signal-pin pattern, and the number of power-pin, as shown in proposed design flow. First of all, we automatically create the pin-block placement sequence via a simple way. As long as we obtain the rough coordinate of each corresponding component, the pin-block placement sequence will be determined by an intuitive manner of enumerating components clockwise (or counterclockwise). Then, we flexibly design and select an appropriate pin pattern as described in Section II-B.

According to the signal-pin name and selected signal-pin pattern, we can automatically construct all signal-pin blocks by locating signal pins within a block along the specific patterns.

The number of power-pin can be used to deal with the power delivery issue. Our strategy is to establish a power-pin block which can provide a power channel on PCB for various power domains. Designers can freely define the demand of power pins for individual signal configuration relying on the power analysis result. While the signal-pin block is constructed, the proposed automation approach will create power-pin block and place it adjacent to the related signal-pin block, then integrate them into single block for a signal bus. Fig. 8 shows an example, nine pin-blocks (#1 to #9) are constructed for nine different interfaces in a package. Finally, the pin-block placement sequence is applied in pin-block grouping strategies which divide all pin-blocks into four group and place on each package side in the next stage, shown in Fig. 8.

We have further developed two strategies for grouping pin-blocks into package boundaries, the boundary-constrained pin-block grouping strategy (BCPG) and the congestion-free pin-block grouping strategy (CFPG). When we design the pin-out for chipset, which acts as a bridge of all components on motherboard, the location of component is one of major constraints presented in Section II-A. Since the locations of components on PCB are boundary-constrained, the grouping strategy BCPG (shown in Fig. 9) will be applied. We defined the safe range for this method

$$\phi_1 \cdot AVG_s \leq S_m \leq \phi_2 \cdot AVG_s \quad (9)$$

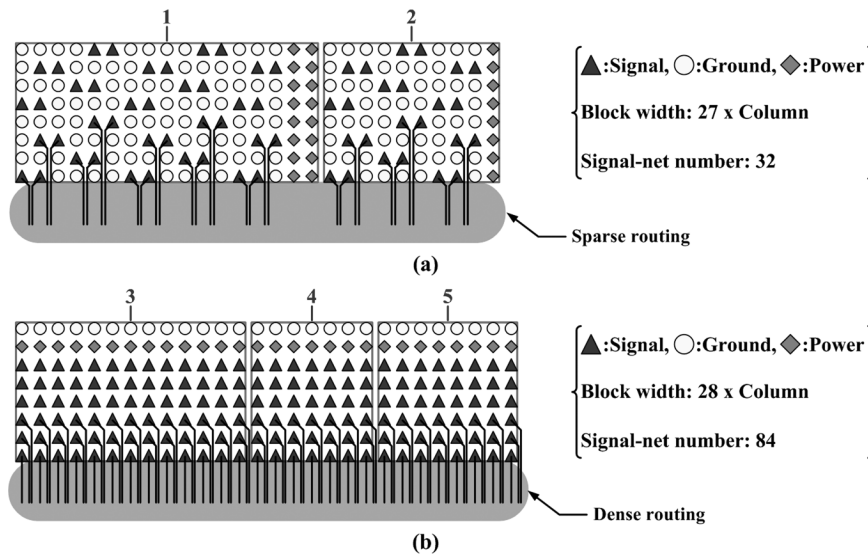


Fig. 9. Boundary-constrained pin-block grouping strategy (BCPG): all pin-blocks will be grouped into single block in proper order until the integrated block size locate within the safe range. The size of grouped pin-block (a) and (b) are closed to each other by adopting BCPG. The disadvantage of this method is that it possibly causes the dense routing likes (b).

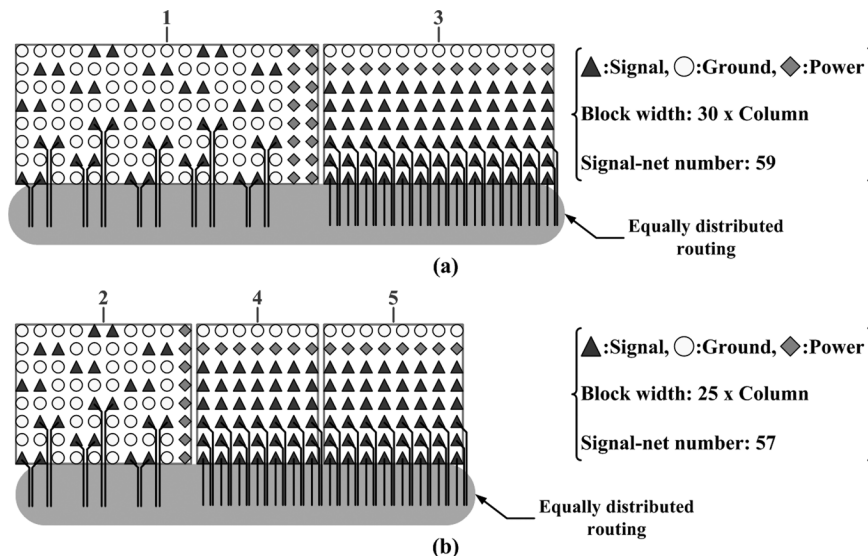


Fig. 10. Congestion-free pin-block grouping strategy (CFPG): the first prior consideration is to equalize the signal-pin number. The signal-pin number of grouped pin-block (a) and (b) are very closed after adopting CFPG. The disadvantage of this method is that the locations of PCB components must be restricted and referred to final pin-out.

where  $S_m$  is the size of grouped block,  $\phi_1$  and  $\phi_2$  are user-defined parameters,  $AVG_s = (\sum_n w_n)/4$  is the average block size and  $w_n$  is the width of each block. Equation (9) shows that the main concern of this methodology is pin-block size. According to the pin-block placement sequence determined in pin configuration, the pin-blocks will be grouped into single block in proper order until the integrated block size locate within the safe range. For this strategy, the size of each grouped block is closed to the average block size then result in minimized  $E_i$  value (shown in Fig. 8) on each side. Therefore, this method will speed up the runtime of minimizing package size. However, the BCPG will introduce the possibility of generating a dense net-routing due to the disregard of signal-pin number. As shown in Fig. 9, the two grouped pin-blocks (one is grouped

with block 1 and 2, the other is grouped with block 3–5) have closed block width, but they have very different signal-net number. The worse case shown in Fig. 9(b) will decrease the routing efficiency on PCB layout and increase the implementation cost for PCB.

The another strategy is CFPG whose primary consideration is to equally distribute signal-pins on each package side. Consequently, the PCB layout will effortlessly lead to a loose density and have more flexibility to match the impedance of critical nets or adjust the location of components. Fig. 10 shows an example, the signal-pin number of integrated blocks will be close to each other when we adopt CFPG strategy. Therefore, this method is suitable for the package design of field-programmable gate array (FPGA) which has the prior concern of routability.

For CFPG, it will consider the signal-pin number instead of the placement order or side of each pin-block, hence the locations of PCB components will be determined after it accomplishes the final pin-out. The safe range used for this strategy is also defined as follows:

$$\psi_1 \cdot AVG_p \leq TP_i \leq \psi_2 \cdot AVG_p \quad (10)$$

where  $TP_i$  is the total signal-pin number of grouped block,  $\psi_1$  and  $\psi_2$  are user-defined parameters,  $AVG_p = (\sum_j p_j)/4$  is the average signal-pin number and  $p_j$  is the signal-pin number of each block. Since the equalized signal-pin number is usually larger than the size of grouped block, the second method must have tighter safe range (e.g.,  $|\psi_1 - \psi_2| < |\phi_1 - \phi_2|$ ) to achieve the same boundary range as that of the first one.

To implement the strategies of BCPG and CFPG, because the pin-block placement sequence will be considered primarily in BCPG, we will use the first-fit heuristic algorithm which is an approximation algorithm for solving bin-packing problem to group pin-blocks. This algorithm sequentially assigns objects into the first bin, and then creates a new bin when the current bin is full. For CFPG, the first prior consideration is to equalize the signal-pin number. Therefore, we can apply another bin-packing approximation, the best-fit heuristic algorithm, to group pin-blocks. This heuristic ignores the order of objects and fills all objects into the feasible bins, which have the smallest residual capacity.

From the observations in Section II, signal integrity, power delivery and routability issues should be accounted for in general cases when signal pins are placed. After finishing the implementation and placement of all blocks, a rough pin designation can be obtained, shown in Fig. 8. At the same time,  $E_1$  to  $E_4$  can be evaluated from this rough pin designation ( $E_1$  to  $E_4$  represent the width or height of the empty and excess area in each side of minimum package). These  $E$  values will be used for package size minimizing and pin-block floorplanning in Section II-B.

### B. Package Size Minimization and Pin-Block Floorplanning

The next step is to optimize package size and acquire a feasible pin designation. The objective function and constraints are formulated as a linear program and shown in the following.

Minimize

$$f = \sum_{j=1,3} \left( \sum_i w_{ji} + E_j \right) h_j + \sum_{j=2,4} \left( \sum_i h_{ji} + E_j \right) w_j$$

subject to

$$W_{\min} = w_4 + \sum_i w_{1i} + E_1 = w_2 + \sum_i w_{3i} + E_3 \quad (11)$$

$$H_{\min} = h_1 + \sum_i h_{2i} + E_2 = h_3 + \sum_i h_{4i} + E_4 \quad (12)$$

$$W_{\min} \geq w_2 + w_4 + w_{\text{Core}} \quad (13)$$

$$H_{\min} \geq h_1 + h_3 + h_{\text{Core}} \quad (14)$$

$$W_{\min} = H_{\min}; w_{\text{Core}} = h_{\text{Core}} \quad (15)$$

$$E_1 + E_2 + E_3 + E_4 \geq 0 \quad (16)$$

where  $w_{1i}, h_{1i}, w_{2i}, h_{2i}, w_{3i}, h_{3i}, w_{4i}, h_{4i}, w_4$  can be evaluated in the previous step, all shown in Fig. 8. The Core is the center area of BGA package. In principal, the power and ground pins are located at the center of package and the die is located upon these power and ground pins. As a result, the heat generated from the die can be transferred out through these pins [1]. Thus increasing more power and ground pins located at the center area will improve heat dissipation but enlarge the area of Core, thereby enlarge the package size. We use (13) to (14) to define the area of Core in accordance with physical die size, where  $w_{\text{Core}}$  and  $h_{\text{Core}}$  are user specified parameters. If these two values are not given by designer the minimum Core size can also be obtained when the minimum package size is evaluated. Constraint (11), (12), and (15) will restrict the shape of package to be square. The purpose of (16) is to insure that the minimum package size can accommodate all pin-blocks with almost no void pin positions.

After  $E_1$  to  $E_4$  are obtained, we can easily recognize the position of the empty and excess area in the minimum package. The final step of proposed methodology is to floorplan pin-blocks, which are to split the pin-blocks in the excess area and fill them into the adjacent empty area. It can completely eliminate exceed area and keep those pins being located around the particular region restricted in previous step. The algorithm of pin-block floorplanning is shown as follows:

- 
- 1)  $i \leftarrow 1, i \in 1, 2, 3, 4$  // start from side 1
  - 2)  $i - 1 \leftarrow 4$ , iff  $i = 1$ ;  $i + 1 \leftarrow 1$ , iff  $i = 4$
  - 3) repeat:
  - 4) while ( $E_i \neq 0 \cap E_i < 0$ ) do
  - 5)   if  $E_{i-1} > E_{i+1}$
  - 6)     shift pins clockwise // fill the pin-block into empty area in last side until the  $E_i$  is zero
  - 7)      $E_i \leftarrow 0, E_{i-1} \leftarrow E_{i-1} + E_i$
  - 8)   else
  - 9)     shift pins counter clockwise // split the pin-block in excess area then group it into next side
  - 10)     $E_i \leftarrow 0, E_{i+1} \leftarrow E_{i+1} + E_i$
  - 11)  $i \leftarrow i + 1$  // check next side
  - 12) until all  $E$  values are large than or equal to zero
- 

Fig. 11 shows an example, where there are two excess areas occurred in second and third side (upper right and upper left corners) and two empty areas occurred in first and fourth side (bottom right and bottom left corners). According to our algorithm, the pin-blocks located in side 1 will be skipped due to  $E_1 > 0$  (line 4 in the above algorithm). While it considers the pin-blocks of side 2 [see Fig. 11(a)], some of the pins in group #3 will be clockwise filled into the empty area in side 1 ( $E_1 > E_3$ , line 5 and 6), and then it will consider the pin-blocks in next side. In side 3 [see Fig. 11(b)], because of  $E_4 > E_2$  the pins of group #7 which excess the range of the side will be split and grouped into side 4 (line 8 and 9). Finally, in the last side the pins of group #9 are the same case as that of group #7 [see Fig. 11(c)] and will be floorplanned into the proper locations then acquire an optimized pin-block floorplanning [see Fig. 11(d)] through this simple procedure.



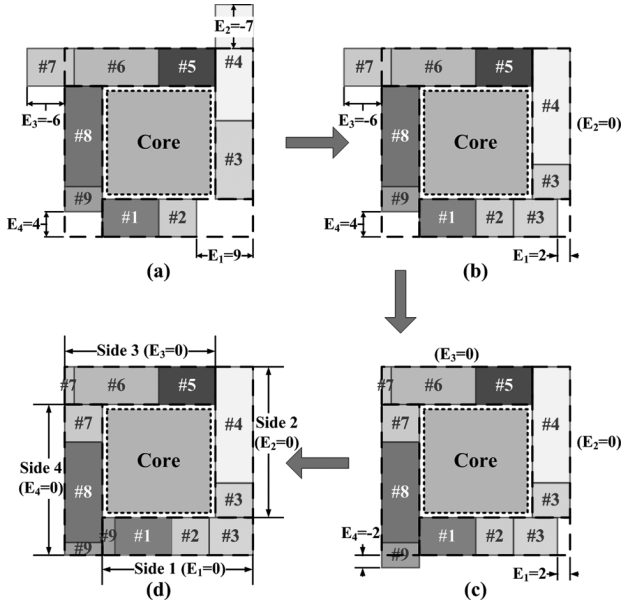


Fig. 11. Example of pin-block floorplanning. The pins in the excess areas will be shifted into the empty areas through our floorplanning algorithm. A final pin-out can be acquired after finishing the package size minimization and pin-block floorplanning.

### C. Dealing With Package Size Migration Issues

For practical application, designer usually need to migrate package size from larger to smaller or vice-versa. During chip prototyping, the extra I/O pins are required for monitoring test signals, and then the package size will be dynamically migrated to a larger one. In addition, when the products have cost margin for improving performance or adding the new features, the chip size and package size must be enlarged simultaneously. These requirements can be satisfied easily by changing types of pin pattern from higher  $SN_i$  (signal-pin number per pattern) to lower one, which increasing the width of pin-blocks. On the contrary, when the cost issue has higher priority than signal integrity or the die size has to be shrunk due to the removal of some features, the package size must be shrunk at the same time. Consequently, the types of pin pattern should be modified from lower  $SN_i$  to higher one, which increased the efficiency of pin designation but relaxed the performance constraints to acquire smaller pin-blocks.

To tackle these package size migration issues, we have defined a migration factor ( $M.F.$ ) to evaluate the enlarged or shrunk column number (width) of pin-blocks during changing types of pin pattern. The migration factor can be simply calculated through the following equation:

$$M.F. = (-1) \cdot col \cdot \left( \frac{1}{SN_p} - \frac{1}{SN_m} \right) \Rightarrow \begin{cases} > 0, & \text{for enlarging package} \\ < 0, & \text{for shrinking package} \end{cases} \quad (17)$$

where  $col$  is the given column number of pin pattern,  $SN_p$  and  $SN_m$  are the signal-pin number per pattern in previous pattern type and modified pattern type. Fig. 12 shows the migration factors of six patterns proposed in this paper, where “+” means

|              | Pattern 1 | Pattern 2, 3 | Pattern 4, 5 | Pattern 6 |
|--------------|-----------|--------------|--------------|-----------|
| Pattern 1    | N/A       | -3/40        | -1/8         | -5/24     |
| Pattern 2, 3 | +3/40     | N/A          | -1/20        | -2/15     |
| Pattern 4, 5 | +1/8      | +1/20        | N/A          | -1/12     |
| Pattern 6    | +5/24     | +2/15        | +1/12        | N/A       |

(M.F. for enlarging pin blocks)

(M.F. for shrinking pin blocks)

Fig. 12. Enlarged and shrunk migration factor ( $M.F.$ ) of six proposed pin patterns, designer can decide the modified pattern along these factors.

TABLE II  
TWO INDUSTRIAL BENCHMARKS USED IN THIS PAPER

|        | Signal bus | Pin number | Order | Selected signal-pin pattern | Power-pin number |
|--------|------------|------------|-------|-----------------------------|------------------|
| Case 1 | Bus #1     | 66         | 1     | 4                           | 32               |
|        | Bus #2     | 27         | 2     | 5                           | 8                |
|        | Bus #3     | 37         | 3     | 6                           | 8                |
|        | Bus #4     | 32         | 4     | 2                           | N/A              |
|        | Bus #5     | 40         | 5     | 2                           | 16               |
|        | Bus #6     | 53         | 6     | 6                           | 16               |
| Case 2 | Bus #1     | 66         | 1     | 4                           | 24               |
|        | Bus #2     | 27         | 2     | 5                           | 8                |
|        | Bus #3     | 95         | 3     | 4                           | N/A              |
|        | Bus #4     | 100        | 4     | 4                           | 8                |
|        | Bus #5     | 42         | 5     | 6                           | 8                |
|        | Bus #6     | 16         | 6     | 6                           | N/A              |

TABLE III  
EXPERIMENTAL RESULTS OF CASE 1 AND CASE 2

|        | $\sum_i E_i$ | Central P/G pins<br>( $w_{Core} \times h_{Core}$ ) | Manually designed package<br>( $W \times H$ ) | Evaluated minimum package<br>( $W \times H$ ) |
|--------|--------------|--|---|---|
| Case 1 | 0            | $10 \times 10$                                     | $26 \times 26$                                | $26 \times 26$                                |
| Case 2 | 3            | $14 \times 14$                                     | $30 \times 30$                                | $31 \times 31$                                |

enlarged factor and “-” means shrunk factor. Therefore, the total pin number of a group multiplied by the migration factor will estimate the modified width of a pin-block. And then designer can decide which pattern should be modified along these estimations.

## IV. EXPERIMENTAL RESULTS

We have implemented our methodology in C++ and the platform is on AMD Sempron 1.75 GHz with 1 GB memory. We use two industrial mass production chipset cases as our benchmarks, the rough pin configuration charts are shown in Table II. Table III, Figs. 14, and 16 show the results of pin-out designation for these two benchmarks. Based on Table III which is obtained from linear programming formulation shown in Section III-B, we can get corresponding parameters to floorplan all pin-blocks. The runtime of designating pin-out is less than 5 s for both cases. For Case 1, when the BCPG strategy is adopted we can obtain very closed pin-block sizes in first and third side [see Fig. 13(a)] and a rough pin-out in minimum package size [see Fig. 14(a)]. Fig. 14(b) shows our

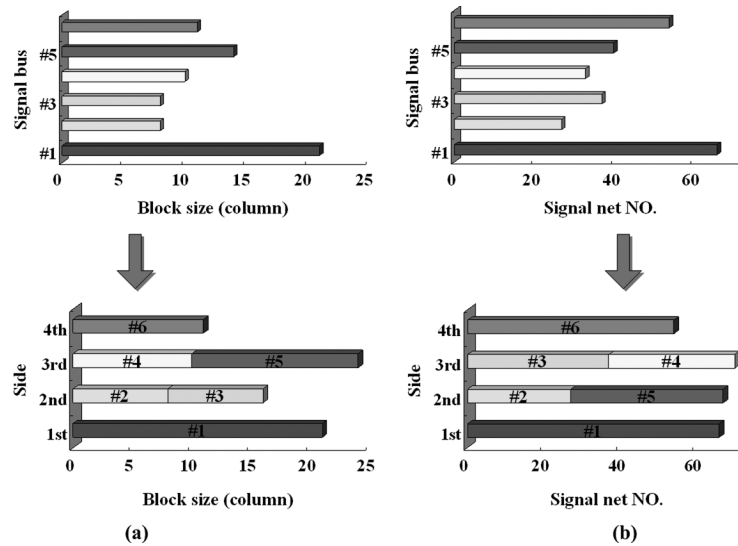


Fig. 13. Pin-blocks grouping results of Case 1: (a) Grouping with BCPG, where the block sizes are very closed in side first and third ( $\phi_1 = 0.5$ ,  $\phi_2 = 1.5$ ). (b) Grouping with CFPG, where the pin number are equalized for all sides ( $\psi_1 = 0.8$ ,  $\psi_2 = 1.2$ ).

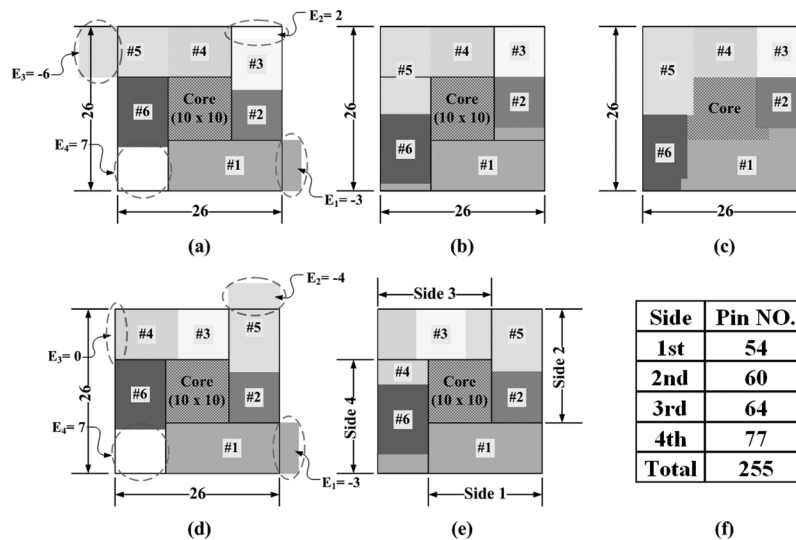


Fig. 14. Experimental results of Case 1: (a) Placement of blocks which are grouped with BCPG in minimum package evaluation. (b) Sketch of pin-out after (a) has been floorplanned. (c) Manually designated pin-out. (d) Placement of blocks which are grouped with CFPG in minimum package evaluation. (e) Sketch of pin-out after (d) has been floorplanned. (f) The equalized signal-pin number chart. By using BCPG to group pin-block, our approach can produce the same dimensions as in manual design. The equalized signal-pin number can also be obtained when CFPG is adopted.

final pin-out designation is perfectly matched with manual design [see Fig. 14(c)] achieved by an experienced engineer, which spent long turn-around time to respin the design (usually weeks). By using CFPG strategy, we can equally distribute signal-pins on each side of package [see Fig. 13(b)], even though the pin-blocks need to be floorplanned [see Fig. 14(e)] the variation of signal-pin number is restricted [see Fig. 14(f)]. For Case 2, the same flow as that in case 1 we choose BCPG strategy to group pin-block first [see Figs. 15(a) and 16(a)]. Due to more pin numbers in some buses and signal-pin block pattern usage [while pin number is not divisible by 8 for pattern (1), 10 for pattern (2) and (3), 12 for pattern (4) and (5), 6 for pattern (6) will generate void position, respectively], a slightly larger package size [see Fig. 16(b)] is achieved, but still very

close to the manual design [see Fig. 16(c)]. Next, we use CFPG method to group pin-block [see Fig. 15(b)], the final pin-out [see Fig. 16(e)] shows the signal-pin number in each side of package is very closed to each other [see Fig. 16(f)]. In addition to these two cases, we have tested a case which has 25 groups and 720 signal-pins, the runtime is still below 5 s and obtain the minimum package size as well.

In package size migration cases, we keep using those two industrial cases as the examples. By (17), we can first calculate the  $M.F.$  as shown in Fig. 12. Then, we multiply pin number in case 1 and case 2 by the  $M.F.$  to obtain the results of Table IV. This table shows the enlarged or shrunk column number of pin-blocks when we are changing types of pin pattern for all groups in case 1 and case 2. For example, if we want to enlarge package

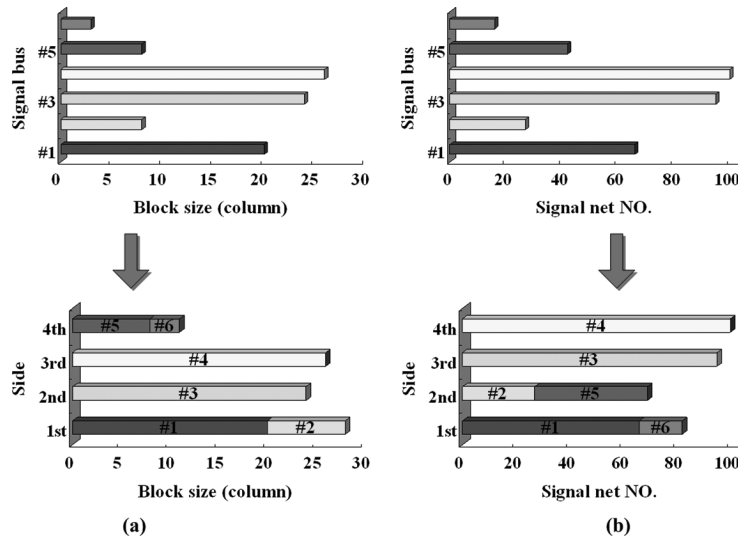


Fig. 15. Pin-blocks grouping results of Case 2: (a) Grouping with BCPG, where the block sizes are very closed in side first, second, and third ( $\phi_1 = 0.5, \phi_2 = 1.5$ ). (b) Grouping with CFPG, where the pin number are equalized for all sides ( $\psi_1 = 0.8, \psi_2 = 1.2$ ).

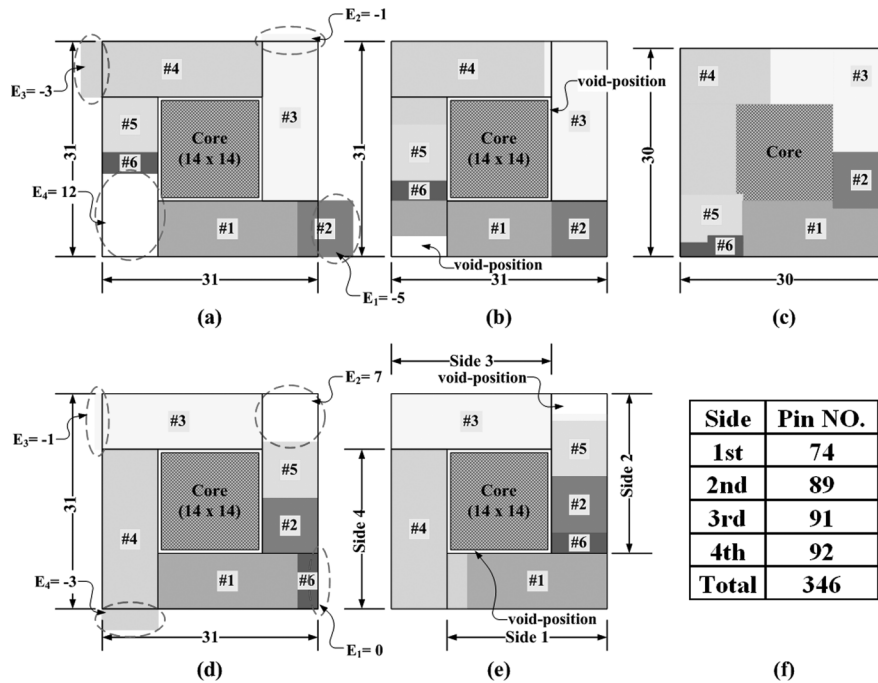


Fig. 16. Experimental results of Case 2: (a) Placement of blocks which are grouped with BCPG in minimum package evaluation. (b) Sketch of pin-out after (a) has been floorplanned. (c) Manually designated pin-out. (d) Placement of blocks which are grouped with CFPG in minimum package evaluation. (e) Sketch of pin-out after (d) has been floorplanned. (f) The equalized signal-pin number chart. When we use BCPG to group pin-block, our approach can produce a pin designation layout ( $31 \times 31$ ) very close to the manual one ( $30 \times 30$ ). The signal-pin number is very closed to each other by using CFPG strategy.

size of case 1 from  $26 \times 26$  (mm) to  $30 \times 30$  (mm), we will need  $(30 - 26) \times 4 = 16$  enlarged column number in pin-blocks. We can change the types of pin pattern in bus #1 to #3 and #6 into first, second, or third pattern, which have good signal integrity to gain better performance. On the other hand, when we are trying to shrink package size of case 2 from  $31 \times 31$  (mm) to  $25 \times 25$  (mm), the shrunk column number will be  $(30 - 26) \times 4 = 16$ . The only one choice is to change types of pin pattern in bus #1 to #4 into sixth pattern. Since the most selected pin patterns have higher pin designation efficiency in this case. The margin

of enlarging package size will be larger than that of shrinking package size.

### V. CONCLUSION

We have proposed a novel and very efficient approach to automating pin-out designation in flip-chip BGA packaging for package-board codesign. Due to the tradeoff in signal performance and package cost, conventional approach usually takes weeks to modify package size and to rework package substrate

TABLE IV  
ENLARGED OR SHRUNK COLUMN NUMBER OF PIN-BLOCKS WITH MODIFIED TYPES OF PIN PATTERN IN CASE1 AND CASE2

|        | signal bus | previous signal-pin pattern | enlarged/shrunk column number of pin-blocks |         |         |         |         |         |
|--------|------------|-----------------------------|---|---------|---------|---------|---------|---------|
|        |            |                             | patt. 1                                     | patt. 2 | patt. 3 | patt. 4 | patt. 5 | patt. 6 |
| Case 1 | Bus #1     | patt. 4                     | +9  | +4      | +4      | N/A     | N/A     | -6      |
|        | Bus #2     | patt. 5                     | +4  | +2      | +2      | N/A     | N/A     | -3      |
|        | Bus #3     | patt. 6                     | +8  | +5      | +5      | +4      | +4      | N/A     |
|        | Bus #4     | patt. 2                     | +3  | N/A     | N/A     | -2      | -2      | -5      |
|        | Bus #5     | patt. 2                     | +3  | N/A     | N/A     | -2      | -2      | -6      |
|        | Bus #6     | patt. 6                     | +12   | +8      | +8      | +5      | +5      | N/A     |
| Case 2 | Bus #1     | patt. 4                     | +9  | +4      | +4      | N/A     | N/A     | -6      |
|        | Bus #2     | patt. 5                     | +4  | +2      | +2      | N/A     | N/A     | -3      |
|        | Bus #3     | patt. 4                     | +12   | +5      | +5      | N/A     | N/A     | -8      |
|        | Bus #4     | patt. 4                     | +13   | +5      | +5      | N/A     | N/A     | -9      |
|        | Bus #5     | patt. 6                     | +9  | +6      | +6      | +4      | +4      | N/A     |
|        | Bus #6     | patt. 6                     | +4  | +3      | +3      | +2      | +2      | N/A     |

and PCB layout, and to rearrange pin-out. Those time-consuming works can be replaced by our efficient methodology. By considering signal integrity, power delivery, and routability in pin-out block design, our framework provides good signal quality while achieving close-to-minimum package size, which reduces package cost. Finally, the flexibility of package size migration will be preserved by a quick and simple estimation.

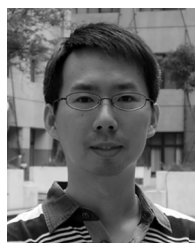
#### ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for providing precious comments that greatly improved this paper.

#### REFERENCES

- [1] Altera Corp., "Designing with high-density BGA packages for Altera devices," Appl. Note AN-114-4.0, Feb. 2006.
- [2] H. N. Brady, "An approach to topological pin assignment," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 3, no. 7, pp. 250–255, Jul. 1984.
- [3] H.-M. Chen, I.-M. Liu, D. Wong, M. Shao, and L.-D. Huang, "I/O clustering in design cost and performance optimization for flip-chip design," in *Proc. IEEE Int. Conf. Comput. Des.*, 2004, pp. 562–567.
- [4] S.-S. Chen, W.-D. Tseng, J.-T. Yan, and S.-J. Chen, "Printed circuit board routing and package layout codesign," in *Proc. IEEE Asia-Pac. Conf. Circuits Syst.*, 2002, pp. 155–158.
- [5] T.-O. Chong, S.-H. Ong, T.-G. Yew, C.-Y. Chung, and R. Sankman, "Low cost flip chip package design concepts for high density I/O," in *Proc. IEEE Electron. Components Technol. Conf.*, 2001, pp. 1140–1143.
- [6] E. Diaz-Alvarez and J. Krusius, "Design, simulation, fabrication, and characterization of package level micro shielding for EMI/EMC management in BGA environment," in *Proc. IEEE Electron. Components Technol. Conf.*, 2000, pp. 793–798.
- [7] S. Hall, G. Hall, and J. McCall, *High-Speed Digital System Design*. New York: Wiley-Interscience Publication, 2000.
- [8] R.-J. Lee, M.-F. Lai, and H.-M. Chen, "Fast flip-chip pin-out designation respin by pin-block design and floorplanning for package-board codesign," in *Proc. IEEE Asia South Pac. Des. Autom. Conf.*, 2007, pp. 804–809.

- [9] J. Mcgrath, "Chip/package co-design: The bridge between chips and systems," *Adv. Packag.*, Jun. 2001 [Online]. Available: [http://ap.pennnet.com/display\\_article/103319/36/ARTCL/none/none/1/Chip/package-co-design/](http://ap.pennnet.com/display_article/103319/36/ARTCL/none/none/1/Chip/package-co-design/).
- [10] N. Oka, C. Miyazaki, T. Uchida, and S. Nitta, "Effect of a shielding plane connected to ground plane of a PCB in EMI reduction," in *Proc. Int. Symp. Electromagn. Compatibility*, 1999, pp. 204–207.
- [11] T. Sudo, Y. Ko, S. Sakaguchi, and T. Tokumaru, "Electromagnetic radiation and simultaneous switching noise in a CMOS device packaging," in *Proc. IEEE Electron. Components Technol. Conf.*, 2000, pp. 781–785.
- [12] J. Xiong, Y.-C. Wong, E. Sarto, and L. He, "Constraint driven I/O planning and placement for chip-package co-design," in *Proc. IEEE Asia South Pac. Des. Autom. Conf.*, 2006, pp. 207–212.
- [13] M.-F. Yu and W.-M. Dai, "Single-layer fanout routing and routability analysis for ball grid arrays," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Des.*, 1995, pp. 581–586.



**Ren-Jie Lee** (S'07) received the M.S. degree in electronics engineering from Feng Chia University, Taichung, Taiwan, in 2000. He is currently pursuing the Ph.D. degree in the Department of Electronics Engineering, National Chiao Tung University, Hsinchu, Taiwan.

From 2000 to 2006, he had worked as a Project Manager for Silicon Integrated Systems (SiS) Corporation, Hsinchu Science Park, Hsinchu, Taiwan. His research interests include beyond die-integration and package/hybrid/board design automation which include Chip-Package-Board codesign, System-in-Package design, and analysis and optimization beyond the die.



**Hung-Ming Chen** (M'04) received the B.S. degree in computer science and information engineering from National Chiao Tung University, Hsinchu, Taiwan, in 1993, and the M.S. and the Ph.D. degrees in computer sciences from University of Texas at Austin, in 1998 and 2003, respectively.

He is currently an Associate Professor with the Department of Electronics Engineering, National Chiao Tung University, Hsinchu, Taiwan. His research interests include EDA (nanometer physical design and design methodology), beyond die-integration, design and analysis of algorithms and optimizations.

Dr. Chen has been a member of some technical committees, including IEEE SOCC, ASP-DAC, and VLSI-DAT.