Selectively Patterned Masks: Structured ASIC with Asymptotically ASIC Performance

Donkyu Baek, Insup Shin, Seungwhun Paik, and Youngsoo Shin
Department of Electrical Engineering, KAIST
Daejeon 305-701, Korea

Abstract— **Structured ASIC, which consists of a homogeneous array of tiles, suffers from large delay and area due to its inherent regularity. A new lithography method called selectively patterned masks (SPM) is proposed. It exploits special masks called masking masks and double exposure technique to allow more than one types of tiles to be patterned on the same wafer. The result is a heterogeneous array of tiles, which relaxes regularity in structured ASIC. A new structured ASIC based on SPM is proposed; tile and routing architectures, design flow, and tile packing and routing algorithm are all addressed. Experiments in 45-nm technology show that, compared to ASIC, the proposed structured ASIC exhibits 2.0 times of area when circuits are optimized for area and 1.2 times of delay when they are optimized for delay. Both figures represent substantial improvement over conventional structured ASIC.**

I. Introduction

A mask cost is one of factors that limit sustaining growth of semiconductor area. A full mask set costs about $300K in 130-nm, $1M in 65-nm, and $4M in 45-nm technology [1]. Gate array is one of solutions to reduce mask cost, in which FEOL (front-end-of-line) layers are pre-fabricated. Its benefit, however, has diminished as the number of BEOL (back-end-of-line) layers increases. FPGA is another option, but the gap between ASIC and FPGA is still too large to consider it as a device for volume production. FPGA is often 10 times slower, takes 50 times more area, and consumes 100 times more power than ASIC counterpart [2], or, more recently, 3.4–4.6 times slower, takes 35 times of area, and consumes 14 times more power [3].

Gate array has developed into structured ASIC during the last 10 years; structured ASIC consists of programmable logics as well as IP cores such as a processor, standard interfaces, and memories, so that both design and manufacturing time can be reduced. Most metal layers are already defined, and programming is performed by customizing contact or via layers, reducing mask cost more than gate array does. A programmable logic consists of an array of identical cells (or called tiles); various tile architectures have been proposed [4]–[6] in consideration of programmability, routability, power consumption, and so on.

Structured ASIC has substantial advantage in mask cost, but it is not yet a device for mainstream use or is not considered as a device that can substitute a standard ASIC. This is mainly due to their inferior performance, e.g. 2.0–6.4 times of delay and 3.0–7.0 times of area [5] compared to ASIC, even though it is much closer to that of ASIC than that of FPGA is. This stems from the fact that structured ASIC inherently imposes too much regularity: it consists of a homogeneous array of tiles and tile architecture has to allow any kind of logic to be implemented on it.

We propose a new concept of lithography called selectively patterned masks (SPM). More than one arrays of tiles are assumed. Each array is patterned on a wafer but selectively, i.e. some tiles are patterned while others are not. This is made possible by using a special mask called a masking mask and double exposure technique during lithography. Selective patterning is necessarily exclusive; if a tile from one array is patterned, a tile from another arrays is not patterned on the same place. The result of SPM is a heterogeneous array of tiles. The motivation of SPM is to relax the regularity of standard structured ASIC, so that its performance can be pushed closer to that of ASIC. The feasibility of SPM in lithography process, in mask design, and in cost model is addressed in Section II.

A prototype of new structured ASIC is presented in Section III. A key component of this prototype is a tile architecture: two types of tiles are proposed to implement combinational gates and one type of tile for flip-flop. Routing architecture, which is another key component since it determines routability and thus circuit delay, is proposed. A design flow from RTL description down to layout is addressed; the flow is built based on commercial CAD tools with customization done through Tcl script, in particular, for tile packing and routing.

The proposed structured ASIC (denoted by SPM) is compared to standard ASIC (denoted by ASIC) in 45-nm technology (Section IV). When circuits are optimized for area, SPM exhibits 2.0 times of area, on average; it exhibits on average of 1.2 times of delay when delay is a target for optimization. These are substantial improvement over conventional structured ASIC. Moreover, there is still a room for improvement, which we briefly address in Section V while we summarize the paper.

II. Selectively Patterned Masks

A. Concept

Suppose that there are three mask sets S_1, S_2, and S_3 as shown in Fig. 1. Each set contains an array of programmable logic, in which programming is done by contacts; a programmable logic shall be called a tile for brevity. A list of logic that can be realized by each set is different, e.g. one tile of S_1 can implement *A*, *B*, or *C* while that of S_2 can implement

978-1-4244-7514-8/11/$26.00 ©2011 IEEE

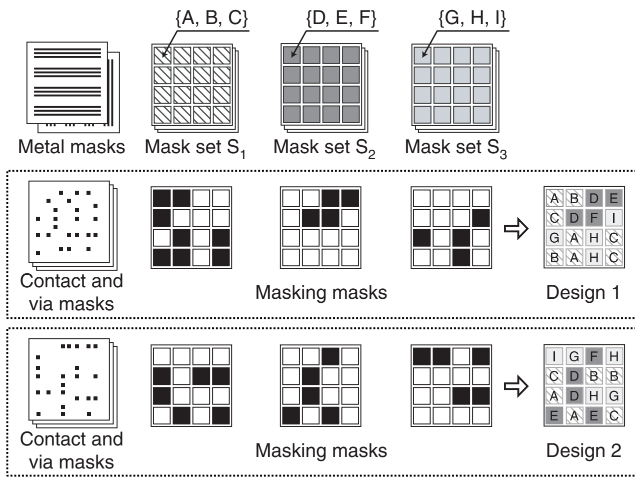

Fig. 1. A concept of SPM and its use in structured ASIC design.

D, E, or F. The connection of tiles is also programmable; programming is done by vias while a mask of each metal layer is pre-fabricated and used in all designs.

Assume that we want Design 1, as illustrated in Fig. 1. To accomplish this design, we have to mix tiles from all three mask sets. This is made possible by introducing special masks, which we call *masking masks*. If each masking mask is used together with corresponding mask set, only the tiles in the set that are not blocked (or masked) appear in the final design; the details of pattern transfer process using masking masks are addressed in Section II-B. As a result, tiles are selectively patterned on a wafer.

A different design, say Design 2 in Fig. 1, which is a different mix of tiles, can be obtained by a new set of masking masks. In this setting, only contact and via masks together with masking masks need to be fabricated for each new design, where the latter can be made with less cost due to their simple geometry. Note that conventional structured ASIC corresponds to a single mask set, which is patterned without masking masks.

B. Pattern Transfer Process

Conceptually, if a pattern mask, a mask from a mask set shown in Fig. 1, and a masking mask are used together (after they are aligned) during lithography process, selective patterning could be achieved. This method, however, is not supported by current lithography equipment. Moreover, the intensity of light will become weaker as it goes through two masks, which makes patterning process more difficult.

The practical solution is to adopt double exposure [7]. Assume that, as shown in Fig. 2(b) after PR strip, a target pattern consists of one polysilicon gate on the left that comes from the first mask set S_1 and two polysilicon gates on the right originating from S_2. As shown in the second step of Fig. 2(a), the photoresist (PR) is first exposed to ultraviolet (UV) light through a masking mask that corresponds to S_1. The PR, in this case, is negative, i.e. the resist that is not exposed to UV light dissolves and disappears. After the second exposure using S_1 mask, which is followed by development, etching, and strip,

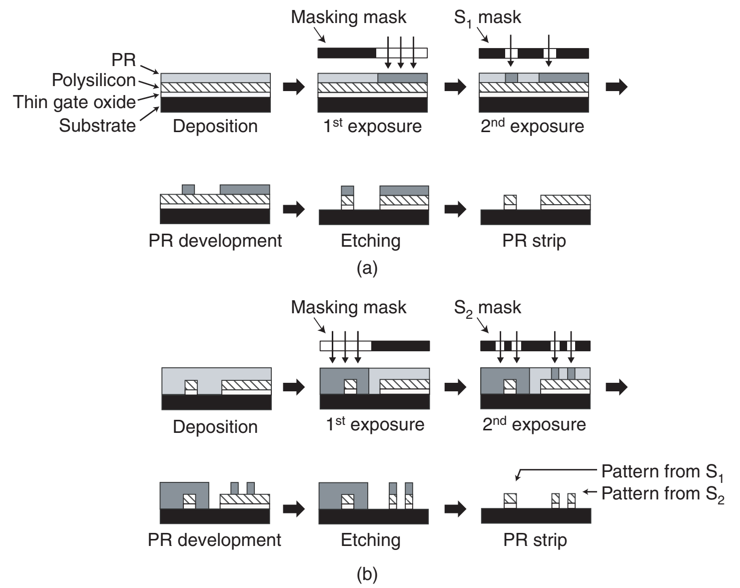

Fig. 2. Pattern transfer process using (a) the first mask set and (b) the second mask set with their corresponding masking masks.

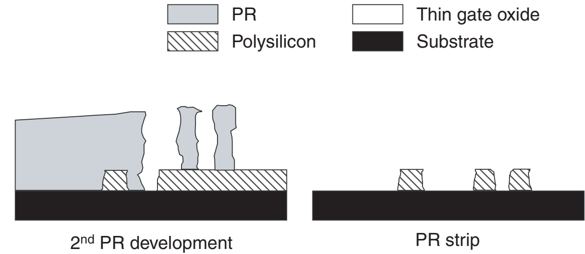

Fig. 3. Lithography simulation of pattern transfer process of Fig. 2.

only the polysilicon pattern on the left remains, even though S_1 mask itself contains two polysilicon patterns. Similar process is repeated using S_2 mask and its corresponding masking mask as shown in Fig. 2(b), which allows two polysilicon patterns on the right to be developed.

The pattern transfer process illustrated in Fig. 2 was tested using a process simulator [8]: PR was set to 800 nm, polysilicon to 250 nm, and thin gate oxide to 3 nm [9]. The result shown in Fig. 3 (corresponding to PR development and PR strip of Fig. 2(b)), where gate oxide is too thin to be observed, confirms that using masking masks indeed allows selective patterning to be realized.

1) Mask Design: A care needs to be taken in designing a masking mask and a pattern mask (recall that a pattern mask is an array of tiles as shown in Fig. 1) to alleviate the effect of light diffraction and potential misalignment between two masks. Fig. 4(a) shows a situation when there is no misalignment between masking and pattern masks; x indicates the minimum distance between a pattern and tile boundary and y corresponds to the amount of masking mask boundary that stretches out beyond tile boundary. Since the light diffracts at the boundary of masking mask, $x+y$ has to be large enough so that the patterns on the left of tile boundary are not exposed to the light (recall that PR is negative). If we assume a light

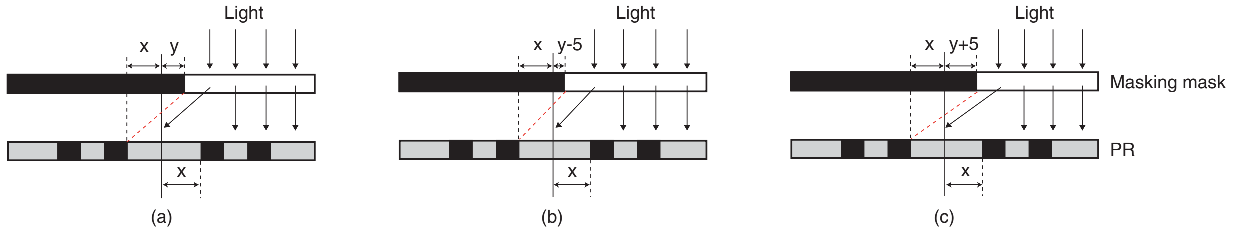

Fig. 4. (a) No misalignment, (b) maximum misalignment to the left, and (c) maximum misalignment to the right.

TABLE I
MASK COST FOR SUB-100NM TECHNOLOGY

Layer	# masks	Cost band	Total cost ($K)
Poly	1	D	40
Active	1	D	40
Deep implants	3	A	24
Drain/source implants	2	C	50
First contact/via	2	E	90
First metal	1	C	25
Other metals	3	B	36
Other vias	2	C	50
Total	15		355

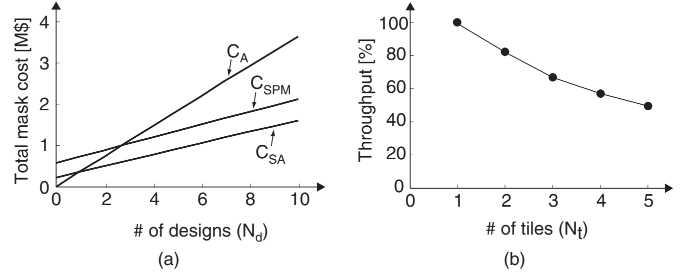

Fig. 5. (a) Total mask cost of ASIC (C_A), conventional structured ASIC (C_{SA}), and SPM (C_{SPM}) and (b) throughput of SPM.

of 193 nm (ArF laser), this corresponds to 80 nm [7]:

$$x+y \geq 80. \tag{1}$$

The patterns on the right of tile boundary has to be exposed to direct light, which yields

$$x \geq y. \tag{2}$$

Fig. 4(b) shows the case when the masking mask is misaligned to the left in the maximum amount. In typical 45 nm technology, this corresponds to 5 nm [10]. The new set of equations therefore becomes $x+y-5 \geq 80$ and $x \geq y-5$. Similarly, when there is a maximum misalignment to the right as shown in Fig. 4(c): $x+y+5 \geq 80$ and $x \geq y+5$. We want to minimize x so that there is less waste of white space in tile design. It can be readily shown that x = 45 nm and y = 40 nm are such quantities, which can be used for designing masking and pattern masks.

C. Cost Model

Table I lists the layers that are typically used in sub-100nm technology [11]. Each layer is associated with the number of masks that are needed during lithography. The cost of mask is denoted by five bands; let the hypothetical cost of bands A, B, C, D, and E be $8K, $12K, $25K, $40K, and $45K, respectively [11].

ASIC involves manufacturing all layers for each new design; its cost, therefore, is given by

$$C_A = 355N_d, \tag{3}$$

where N_d is the number of designs. In conventional structured ASIC, only contact and via layers are newly manufactured; all other layers have to be prepared a priori, which constitute the initial investment. Its cost model, therefore, is

$$C_{SA} = 215+140N_d. \tag{4}$$

In SPM, poly, active, drain/source implants, and first/second metal layers have to be prepared for each mask set, while deep implant and all other metal layers are shared (see Fig. 1); the initial investment, therefore, is $48+167N_t$, where N_t is the number of different tiles. A masking mask can be made with less cost due to its simple geometry; its cost is assumed to be in band A. The cost of SPM can be modeled by

$$C_{SPM} = 48+167N_t+8N_tN_d+140N_d. \tag{5}$$

Fig. 5(a) plots (3), (4), and (5), while we fix N_t to 3. Note that the slopes of C_{SPM} and C_{SA} are not very different (164 versus 140), which suggests that the structured ASIC using SPM closely follows the cost model of standard structured ASIC. As there are more number of new designs (N_d), the cost difference between ASIC and structured ASIC becomes apparent, as it must.

Manufacturing time increases in SPM, which degrades throughput. A rough estimate of throughput is modeled by

$$\text{Throughput} = \frac{T_A}{T_A+24.7N_t}, \tag{6}$$

where T_A is the manufacturing time for ASIC. Fig. 5(b) plots the throughput while N_t is varied; the throughput degrades by 33% when $N_t = 3$.

III. STRUCTURED ASIC DESIGN WITH SPM

A. Tile Architecture

To assess SPM toward structured ASIC design, we have constructed three different tiles, each one belonging to its own mask set as shown in Fig. 1. Two of them, which are illustrated in Fig. 6(a) and (b), are used to implement combinational logic gates and the remaining one, shown in Fig. 6(c), is used to realize a flip-flop; they shall be denoted by T_1, T_2, and T_3, for brevity.

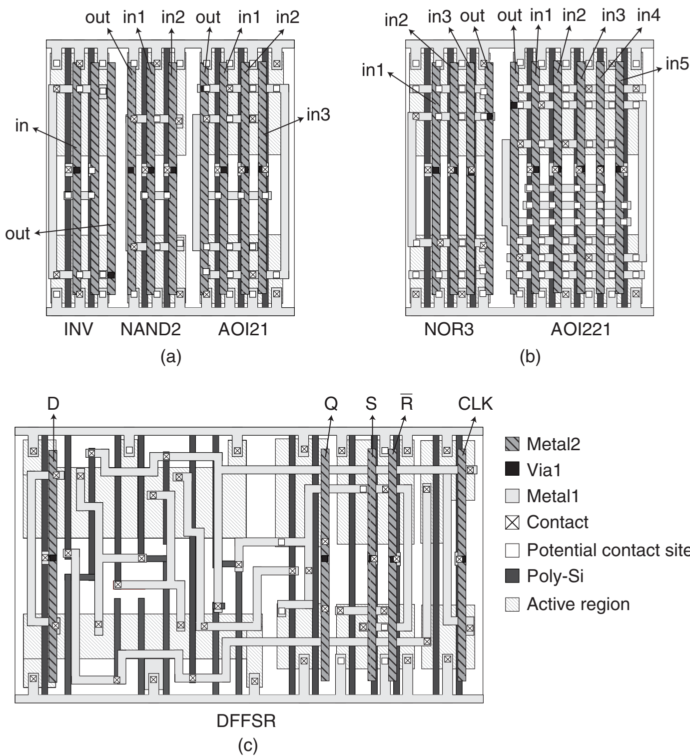

Fig. 6. Architecture of three tiles; an example of programming to implement (a) INV, NAND2, and AOI21, (b) NOR3 and AOI221, and (c) set-reset D flip-flop.

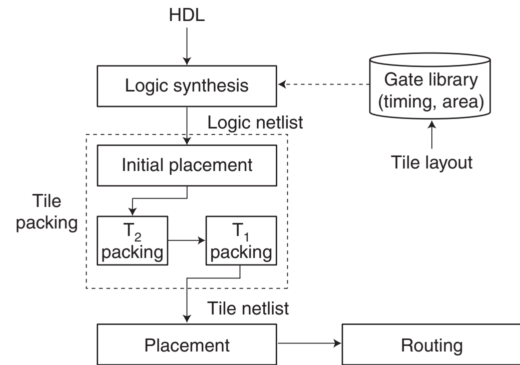

Fig. 7. Overall flow of structured ASIC design with SPM.

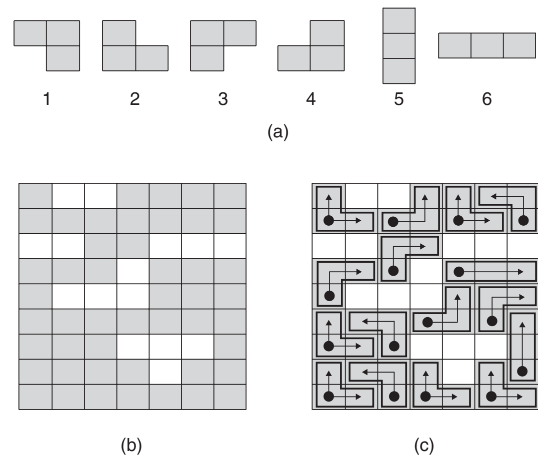

Fig. 8. T_1 packing: (a) packing patterns with their priorities, (b) an example before packing, and (c) a result of packing.

The width of T_1 and T_2 is a half of that of T_3. All the tiles have the same height, which is determined in consideration of routability. There are 10 pins (see Metal2) in each of T_1 and T_2. During routing, which is addressed in Section III-C, a connection can be made to each pin using horizontal Metal3 track. By allowing one more track for a connection that simply goes through the tile, the height consists of 11 Metal3 tracks. There are 5 pins in T_3; the remaining 6 tracks, therefore, can be used for feed-through connection.

The tile T_1 has been designed to accommodate three logic gates: two of them can implement any logic gates consisting of up to two polysilicon gates (see Fig. 6(a)) such as INV and NAND2; the remaining one implements a logic gate that requires three polysilicon gates including AOI21. Two logic gates can be implemented in T_2 as shown in Fig. 6(b); complex gates such as AOI221 that requires five polysilicon gates can be realized on the right portion of the tile. Note that T_1 and T_2 support gate sizing to some extent, e.g. 2× INV can be implemented where 2 polysilicon gates are available and 4× INV can be realized on the right of T_2 by using 4 polysilicon gates. Four types of flip-flops, depending on whether set or reset inputs are used, can be implemented in T_3.

B. Design Flow

The overall design flow based on the proposed structured ASIC is illustrated in Fig. 7. An RTL design written in HDL is given to a commercial logic synthesis tool [12]. Each gate in the layout (see Fig. 6) was simulated using SPICE to extract its timing information (delay and output transition time, each as a function of load capacitance and input transition time). An area was extracted assuming that a gate is implemented in minimum area and a tile is fully utilized, e.g. the area of INV was assumed to be the area occupied by INV implementation as shown in Fig. 6(a) even though it can be implemented anywhere in T_1 or T_2. A library was built from the extracted timing and area information, and was used during logic synthesis. In the output of logic synthesis, however, each gate is assumed to occupy a whole tile, i.e. one tile consists of only one gate, because packing of more than one gates into T_1 or T_2 tiles has not been performed yet. The corresponding netlist is called a logic netlist as opposed to a tile netlist.

Packing is then performed on the logic netlist. A simple heuristic was developed for this purpose, which is depicted by the dotted box in Fig. 7. An initial placement is performed [13] using a logic netlist. A tile, corresponding to a single gate at the moment, is denoted by T_1 if that gate can be implemented by either T_1 or T_2 such as NOR3; it is denoted by T_2 if T_2 is the only tile that can implement the gate such as AOI221 (see Fig. 6). The proportion of T_2 is typically small. We thus perform T_2 packing first; each T_2 tile is picked one by one and combined with any adjacent T_1 tile.

T_1 packing can be explained using an example shown in

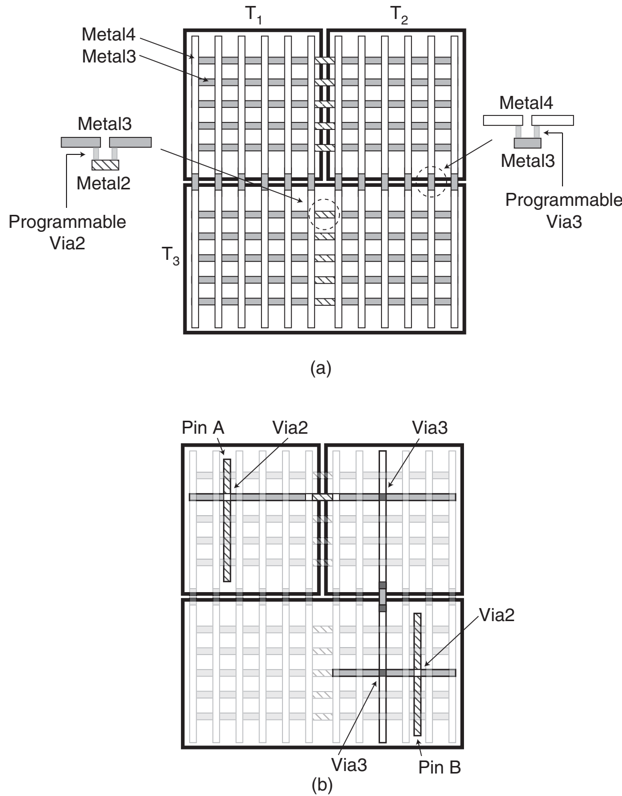

Fig. 9. (a) Routing architecture and (b) an example of connecting pins.

Fig. 8(b). Let each shaded box be T_1; each white box may correspond to T_2, which is already packed, or two consecutive white boxes correspond to T_3. The objective is to group three adjacent T_1s into a single T_1 tile. This is done by Tetris-like packing; we visit each shaded box from the bottom left toward top right corner, and try to find a packing pattern (in order of pattern priority) that fits in the current box as illustrated in Fig. 8(a); Fig. 8(c) shows a result.

Final placement is then performed on the result of packing, corresponding to a tile netlist, which is followed by routing.

C. Routing

1) Routing Architecture: Routing is performed by programming vias; metal layers are fixed and used in all designs (see Fig. 1) so that their masks can be shared. Fig. 9(a) shows a routing architecture. One tile of T_1 or T_2 is overlaid with a grid made of 11 horizontal M3 tracks and 12 vertical M4 tracks (Fig. 9(a) is drawn as a simplified form); two grids are thus used by a single T_3 tile. To make a connection between M3 tracks in adjacent grids, an array of M2 segments (hatched patterns) is used; the real connection is made by programming vias. Similarly, an array of M3 segments is used for a connection between M4 tracks in adjacent grids.

Fig. 9(b) illustrates how two pins in different tiles can be connected. Pin A is located in M2 layer (see Fig. 6); it is brought up to M3 using a via; the connection to pin B is then made through M2 segment, M3 segment, and several vias. A heavy use of vias and dangling wire segments may increase gate- and wire-delay, which we analyze in Section IV-B.

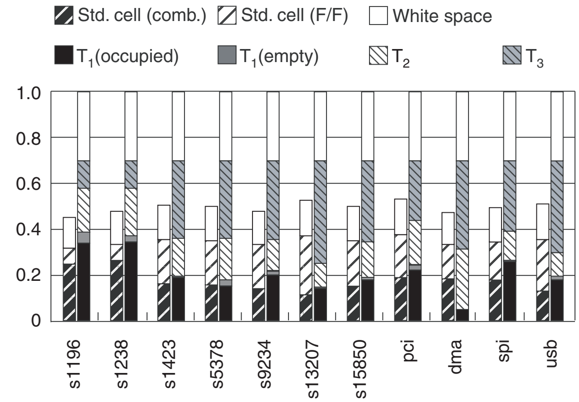

Fig. 10. Normalized area of ASIC (left bar) and SPM (right bar).

2) Routing Algorithm: A simple routing method was developed using a Tcl script executing on a commercial routing package [13]. An ASIC-style trial routing is first performed on a tile netlist without assuming the proposed routing architecture. The top K critical paths are selected; each path is considered in the order, and each net on the path is routed one by one following the method shown in Fig. 9(b). The remaining nets are then routed in decreasing order of their wirelength. A router specific to the proposed routing architecture is left for a future development.

IV. Experimental Results

Experiments were carried out to compare the proposed structured ASIC (denoted by SPM) with standard ASIC (denoted by ASIC) in area and delay; 7 sequential circuits were taken from the ISCAS benchmarks and 4 were taken from open cores [14]. In ASIC, each circuit was synthesized [12] with a gate library [15] consisting of 134 gates, which is based on 45-nm technology; the same circuit was synthesized in SPM, in which a total of 31 gates are available (see Fig. 6). The design flow for SPM, illustrated in Fig. 7, is based on commercial tools with customization (such as packing and routing) done by Tcl script.

A. Assessment of Area

Logic synthesis was performed to minimize area (without regard to delay) in both ASIC and SPM. We forced about 70% of the placement region to be occupied by cells or tiles during physical design, which is a tight placement. Fig. 10 shows a result, in which all figures are normalized to the total area of circuit from SPM.

The circuits from SPM occupies, on average, 2.01× of the area of circuits from ASIC. There are several factors that contribute to the area increase. The height of tile (see Fig. 6) is 49% larger than that of standard cell; this is the result of effort to increase routability through horizontal M3 tracks (see Fig. 9) so that limited routing due to regular routing architecture (as opposed to random routing in ASIC) can be alleviated. Another factor is under-utilization of tiles: INV uses only one polysilicon even though twos are available (see Fig. 6(a)), thus one is wasted; packing method presented

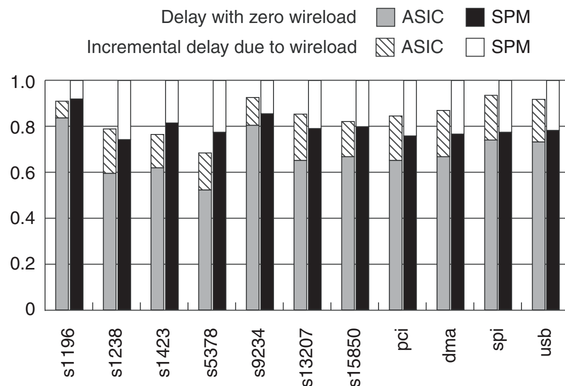

Fig. 11. Normalized delay of ASIC (left bar) and SPM (right bar).

in Section III-B is not perfect leaving some tiles not fully occupied. Packing, however, turned out to contribute very little in area increase. As shown in Fig. 10, T_2 tiles were always packed and very small proportion of T_1 tiles were left unpacked, which yielded on average of 4% of total tile area being empty.

The proposed method, however, has advantage when compared to conventional structured ASIC: 3×–7× [5] and 4× of area [6] compared to ASIC design.

B. Assessment of Delay

Logic synthesis was performed to minimize delay this time; zero wireload was assumed during the synthesis. ASIC design was then followed by placement and routing, both of which are timing-driven; wireload was then extracted for post-layout timing analysis. Routing algorithm presented in Section III-C.2 was applied to SPM design. Fig. 11 shows the result, where the lower portion of each bar corresponds to the delay after logic synthesis (thus 0 wireload) and the height of whole bar denotes the delay after layout.

The delay of SPM design is 1.19× of that of ASIC design, on average, which is very promising; conventional structured ASIC is reported to be slower than ASIC by 2.0×–6.4× [5]. Three circuits (s1238, s1423, and s5378) are rather slow when implemented with SPM. The lack of library gates (31 gates) seems to be the main reason since they are slow even before physical design. The routing architecture illustrated in Fig. 9 does not affect the delay too much because of timing-driven routing. The wires on critical path will be maintained short even in the proposed architecture.

C. Design Space: Area Versus Delay

Fig. 12 shows a design space spanned by area and delay for s1238. Note that Fig. 10 and Fig. 11 correspond to the rightmost and leftmost data points, respectively.

Note that SPM exhibits 2.1× of area when the circuit is optimized for area, but 1.69× under delay optimization; similarly, there is 1.26× of delay difference when the circuit is optimized for delay, but 0.9× under area optimization. This can be partly understood from the number of library gates (31 in SPM and 134 in ASIC) and degree of regularity (regular in SPM and random in ASIC). For instance, when ASIC optimizes a circuit for area, it does a better job than SPM, which yields 2.1× of difference. When ASIC optimizes a circuit for delay, however, area is easily sacrificed while it is not much in SPM due to regularity and lack of library gates, which yields 1.69× of difference.

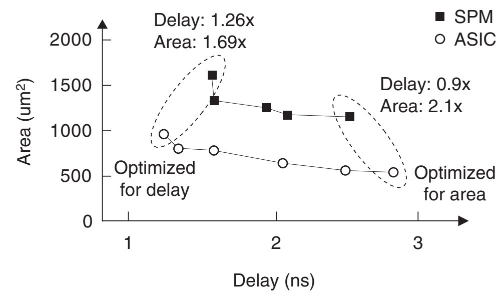

Fig. 12. Area versus delay curve of s1238.

V. Conclusion

We have presented a new concept called SPM, which allows more than one type of tiles to be used in structured ASIC design. The motivation of SPM is to relax too much regularity in structured ASIC so that its performance (area and delay) is pushed closer to ASIC design, while the cost model remains intact. A prototype of new structured ASIC has been developed, addressing tile architecture and design flow. Experiments show 2× of area and 1.2× of delay compared to ASIC design, which are substantial improvement over conventional structured ASIC.

Acknowledgment

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2010-0013439).

References

[1] M. Ooishi, "TSMC takes lead in 45nm IC mass production," *Nikkei Electronics Asia*, July 2007.
[2] P. Zuchowski et al., "A hybrid ASIC and FPGA architecture," in *Proc. ICCAD*, Nov. 2002, pp. 187–194.
[3] I. Kuon and J. Rose, "Measuring the gap between FPGAs and ASICs," *IEEE Trans. on CAD*, vol. 26, no. 2, pp. 203–215, Feb. 2007.
[4] L. Pileggi et al., "Exploring regular fabrics to optimize the performance-cost trade-off," in *Proc. DAC.*, June 2003, pp. 782–787.
[5] N. V. Shenoy, J. Kawa, and R. Camposano, "Design automation for mask programmable fabrics," in *Proc. DAC.*, June 2004, pp. 192–197.
[6] Y. Ran and M. Marek-Sadowska, "An integrated design flow for a via-configurable gate array," in *Proc. ICCAD*, Oct. 2004, pp. 582–589.
[7] M. Fritze et al., "High-throughput hybrid optical maskless lithogrphy: all-optical 32-nm node imaging," in *Proc. SPIE*, Mar. 2005, pp. 2743–2748.
[8] Silvaco, "Athena User's Manual," Dec. 2002.
[9] S. Yang et al., "A high performance 180nm generation logic technology," in *Proc. IEDM*, Dec. 1998, pp. 197–200.
[10] W. Arnold, M. Dusa, and J. Finders, "Manufacturing challenges in double patterning lithography," in *Proc. Int. Symp. on Semiconductor Manufacturing*, Sept. 2005, pp. 283–286.
[11] A. Balasinski, "Mask cost for sub-100nm technologies: stopping a runaway?," in *Proc. SPIE*, June 2003, pp. 82–92.
[12] Synopsys, "Design Compiler User Guide," Sept. 2008.
[13] Cadence, "SoC Encounter User Guide," Nov. 2007.
[14] "OpenCores benchmarks," http://www.opencores.org/.
[15] "Nangate cell library," http://www.nangate.com/.