ASIC Design Methodology with On-Demand Library Generation

Hidetoshi Onodera, Masanori Hashimoto, and Tetsutaro Hashimoto
Department of Communications and Computer Engineering, Kyoto University

Abstract

This paper describes a custom design method of ASICs with on-demand library generation. According to the result of performance estimation, a tailored library is generated and supplied to cell-based design tools. A symbolic layout system that produces a cell layout with variable driving strength is developed. The tunability can be utilized for generating a rich set of driving strength as well as design optimization in post-layout stage. Design experiments and measured performance of a fabricated chip demonstrate the effectiveness of the proposed approach.

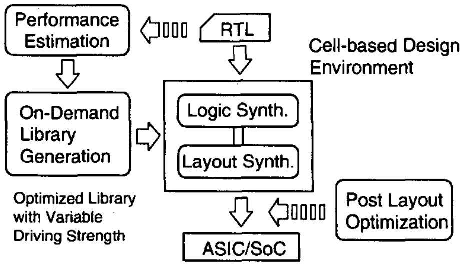

Performance Estimation
RTL
Cell-based Design Environment
On-Demand Library Generation
Logic Synth.
Layout Synth.
Optimized Library with Variable Driving Strength
Post Layout Optimization
ASIC/SoC

Fig. 1. ASIC design methodology with on-demand library generation.

Introduction

Current ASIC design methodology relies on pre-designed libraries. The least abstracted design entity is a logic gate in a library. In general, a few sets of libraries are provided by silicon foundry or library vendors for each fabrication process, and shared with all the ASICs designed for the process. This cell-based design methodology is well-established and enables to design over a million gate ASICs (hopefully) within a reasonable design time and effort. However, target specifications of ASICs are inherently diverse, and therefore a few sets of fixed libraries cannot provide an optimal solution for each ASIC. Also, as the technology goes into an ultra deep sub-micron regime, loading condition of each gate inside a chip may vary very much due to unpredictable wire loading, which would make timing closure difficult. There exists a wide performance gap between ASICs and custom ICs [1]. The use of fixed libraries should responsible in part for the performance loss.

The use of automatic generated libraries for processor ICs is reported[2]. The concept of tailored libraries should also be beneficial for ASIC design. This paper describes a design methodology of optimized ASICs that utilizes libraries generated on demand for each ASIC. The objective of this methodology is to achieve full custom performance with the cell-based design framework for ASICs. Existing tools for logic and layout synthesis with additional post-processing can design highly optimized ASICs with reasonable design effort.

This paper is organized as follows. We first explain the proposed design methodology. Key issue is the on-demand library generation which will be described next. The result of design optimization using conventional ASIC design tools is demonstrated, together with the result of a real chip example. A chip can be further optimized in a post-layout step. It's method and the result will be discussed. Finally we will conclude our discussion.

ASIC design with On-Demand Library Generation

Key concept of the proposed design methodology is to exploit existing cell-based design tools for ASICs for delivering full custom performance. Advanced design method such as library-less synthesis[3] which is not compatible with current cell-based design framework is not considered here. Fig. 1 explains the proposed design methodology. An ASIC is designed from an RTL description using conventional design tools. The main difference from a conventional flow is the generation of library tailored for each ASIC. According to the result of performance estimation, a set of cell libraries with variable driving strength are generated. For conventional design tools, the tunability of driving strength is used to provide a library with a rich set of fixed driving strength. After the completion of layout, this tunability is fully exploited in the post-layout optimization process depicted in the right side of Fig. 1. This optimization process is another feature of the proposed methodology that improves power dissipation and reliability while keeping the operating speed unchanged.

On-Demand Library Generation

We generate a cell library suitable for each ASIC and supplies it to cell-based design tools. In the library generation, we should consider the varieties in functionality and driving strength that should be covered by the library. We have examined the effect of the functionality and driving strength experimentally using a commercial logic and layout synthesis tools. Our conclusion is that the varieties in functionality are not necessarily to be large. On the other hand, the range of driving strength should be wide enough to optimally meet various loading conditions. Based on this observation, we determine the functionality of our library as listed in TABLE I. They are relatively simple compared with commercial libraries, but lead to comparable performances. For each logic in the table, we should prepare a wide variety of driving strength optimized for each application.

We have developed a layout generation system called VARDS that can produce a cell layout with variable driving strength[4]. Transistor-level automatic P&R methods for cell layout generation are reported[5]. They may be used for the on-demand generation of layout. However, the performance of resulting layout should be predictable in order to have a high-level of confidence in quality. Also, an incremental change of driving strength should be possible while keeping other characteristics unchanged. These requirements can be best met by layout generation based on pre-defined symbolic layout.

The layout generation system should have the following features.

- It should be process-portable.
- Layout should be dense.
- Each transistor-width can be varied while the cell height is unchanged.
- The location of each pin should be fixed while transistor widths are changed.

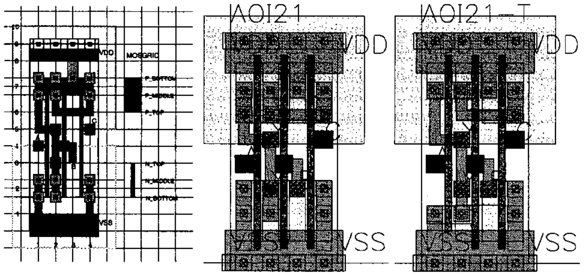
VARDS meets these requirements by a symbolic layout method using a hierarchically defined virtual grid[4]. Given a set of design rules and constraints on resulting layout such as cell height and rail width together with each transistor size, VARDS produces a real layout from a symbolic layout. This process is shown in Fig. 2. Fig. 2(a) is a graphical view of a symbolic layout (AOI21). Fig. 2(b) is a generated layout with maximum width for each transistor, while Fig. 2(c) has different size for each transistor. Thus we can tune(reduce) driving strength while keeping the pin location unchanged. This property is very important for post-layout performance optimization on the design after the completion of derailed routing. If we need a stronger driving strength than that of Fig. 2(b), VARDS can enlarge transistor width of all the cells in a library to produce a cell with larger cell height. In case we need a strong cell with the same cell height, we should change the symbolic layout with parallel transistors or multiple-stage(buffered) cells prepared beforehand.

The quality of cell layout generated by VARDS is evaluated under four different technologies. TABLE II lists area penalty and performance loss compared with hand-crafted cells. The rightmost column indicates the minimum transistor width with respect to its standard(maximum) width. The cell height is 9 routing pitches. The reduction in transistor width does not reduce layout area because the cell height is fixed. However, it contributes to the reduction in power dissipation as shown in a later section. From the result, the size of the cells by VARDS is about 7 % larger but the performance overhead is negligible, while its driving strength can be tuned from 100 %(the maximum value) down to about 30%.

For cell-based design environment, delay and power dissipation of each cell should be characterized beforehand. Two dimensional look-up tables with respect to output loading and input transition rate are commonly used for conventional design tools. These tables can be prepared by an automatic characterization system using circuit simulation[6]. The reduction in computational cost consumed by circuit simulation can be reduced by the use of a quasi-analytical approach that derives delay and power dissipation with the minimum use of circuit simulation[7]. Currently about 40x speed up is obtained for the generation of two dimensional look-up tables.

TABLE I
LIST OF PRIMITIVE GATES

INV, BUF, TBUF
NAND2, NAND3, NAND4, AND2, AND3, AND4
NOR2, NOR3, NOR4, OR2, OR3, OR4
AOI21, AOI211, AOI22, AO21, AO211, AO22
OAI21, OAI211, OAI22, OA21, OA211, OA22
XOR2, XNOR2, FAD, HAD
DF, DFR, DFSR



(a) symbolic layout (b) max. width (c)different width

Fig. 2. Layout generation with variable driving strength from a symbolic layout.

Design Examples

The effect of the proposed design methodology, before the application of post layout optimization, is examined through design experiments. Also, the methodology is applied to a real circuit and the performance is measured on a fabricated chip.

A 32-bit RISC processor core[8] is designed under three different design specifications(clock frequency) of 100 MHz, 120 MHz, and 130 MHz. A target process is a 0.35 μm with three metal layers. For comparison, the circuit is also designed with a fixed library which is used for actual fabrication.

The performance estimation for the on-demand library generation is performed in the following manner. First, the statistics of the circuit such as maximum logic depth, the average number of fan-outs, the average amount of wire loading, are obtained by an initial synthesis and place and route. Then a critical-path model is created which is a chain of NAND2 with the average wire and fan-out loading, the depth of the chain being equal to the maximum logic depth. The delay of the

TABLE II
LAYOUT QUALITY(AREA PENALTY, PERFORMANCE LOSS, MINIMUM TRANSISTOR WIDTH)

Process	Area	Delay	Power	Min. width
A(0.35μm)	7.1%	1.1%	0.8%	27%
B(0.35μm)	7.1%	1.1%	0.8%	22%
C(0.5μm)	2.1%	0.3%	0.2%	35%
D(0.6μm)	2.4%	--	2.7%	32%

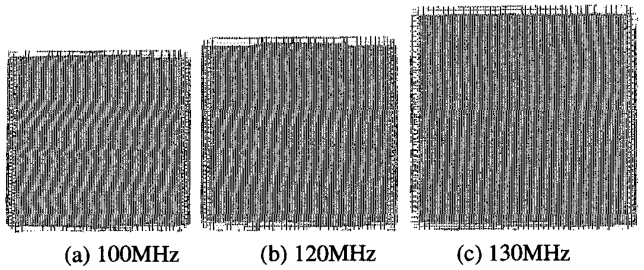


Fig. 3. 32 bit RISC processor designed with a library generated on-demand.

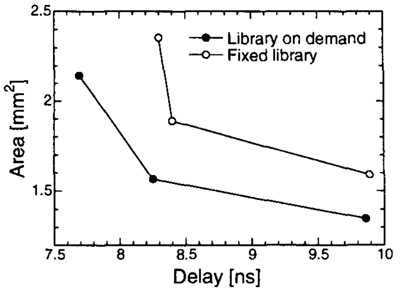


Fig. 4. Area-delay tradeoff characteristics.

critical-path model as a function of transistor size is evaluated by circuit simulation. From the delay versus transistor width characteristic, we can determine the required width for a given specification on circuit delay.

Based on the estimation, we generate libraries with 9 routing-pitch height, 10 pitch height, and 12 pitch height for 100MHz, 120 MHz, and 130 MHz specifications, respectively. The 10-pitch library and the 12-pitch library are later replaced with a 11-pitch library and a 13-pitch library respectively during logic synthesis stage, because timing closure is predicted to be difficult with the initial libraries. A commercial logic synthesis tool and a layout synthesis tool are used for the experiment. At the early stage of the design where design uncertainties are large, we only use a course set of varieties in driving strength such as x1, x2, x4, x8, etc. At the late stage of the design after the placement of each cell is obtained, we add intermediate varieties such as x0.5, x0.75, x1.5, x3, etc., which are used for power optimization.

According to the proposed design methodology with the on-demand library generation, we can design final layouts which meet the given specifications of 100MHz, 120MHz, and 130MHz, respectively, as shown in Fig. 3. On the other hand, the circuits with the fixed library meet the specification of 100MHz, but fail to meet those of 120MHz and 130 MHz. The area-delay tradeoff curves for both cases are shown in Fig. 4. As for the circuits under the 100MHz specification, the area and the power dissipation are 15 % and 29 % smaller in the circuit with the library generated on-demand.

Next we show a real chip example. The circuit is a DSP chip for moving picture compression with 160 bit buses[9]. For comparison, two cores are designed and integrated on the same die; one with the library generated on-demand and the other with the fixed library which is the same library used for the above experiments. The chip micro-photograph is shown in Fig. 5, the top portion is the core with the fixed library and the bottom is the core with the library generated on-demand. Some statistics of the cores are listed in TABLE III. The core area is the size of the core while the cell area is the total area of all the cells in the core. There exists a large difference between them, which indicates the core is routing-resource limited due to the huge width of data buses. This routing dominated chip is not a good circuit for evaluating the methodology. Nevertheless, the core designed by the proposed methodology has about 9 % less area and 9 % less power dissipation measured at 1.6 V.

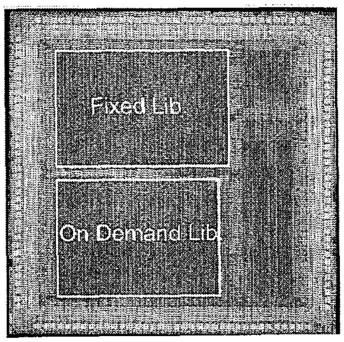


Fig. 5. Micro-photograph of a fabricated DSP chip.

Post Layout Performance Optimization

An important feature of the library generated on demand by VARDS is the ability of tuning(down-sizing) its driving strength without changing the location of input/output pins. This ability enables to optimize the circuit after detailed routing where exact amount of wire loading and crosstalk noise characteristics can be extracted. We can optimize each transistor size(driving strength) while preserving interconnect structure after detailed routing. This does not contribute to area reduction but improves power dissipation and cross-talk noise performance. A conventional cell-based design tool does not support transistor sizing, thus we have developed a dedicated tool for the post-layout optimization. A heuristic algorithm for transistor sizing based on sensitivity calculation is devised[10]. Cell delay is calculated from four-dimensional look-up tables with respect to output loading, input transition rate, pull-up transistor width, and pull-down transistor width. During the

TABLE III
STATISTICS OF DSP CORES

	On-Demand	Fixed
Core area(mm^2)	4.26	4.68
Cell area(mm^2)	1.78	2.15
# Cells	15,324	13,466
Measured Pd. @25MHz, 1.6V	37mW	41mW

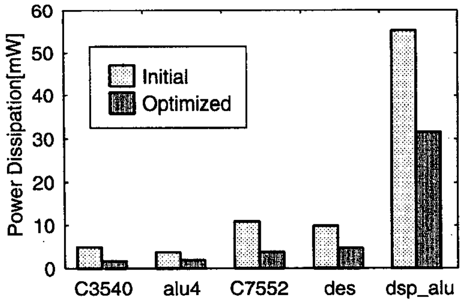


Fig. 6. Results of post-layout optimization for power reduction.

optimization, noise margin of each cell is maintained above a pre-specified value such as 1/4 Vdd. Also, signal transition rate is controlled within a specified value. The latter constraint is important for maintaining the accuracy in timing analysis and improving hot-carrier reliability.

The effect of the post layout optimization is evaluated experimentally. The circuits used for the experiments are an ALU in the DSP shown in Fig. 5 (dsp_alu with Vdd of 3.3 V) and the circuits included in ISCAS85 and LGSynth93 benchmark sets (C3540, alu4, C7552, des). An initial library with the height of 9 routing-pitches is generated by VARDS assuming a 0.35 μm technology. The library has six varieties in driving strength for INV and BUF. Other cells have four varieties. Initial circuits are synthesized under two different constraints; minimizing circuit delay and minimizing circuit area. A constraint on the transition time of 0.5 ns is imposed. Wire capacitance values are extracted from layouts after derailed routing, and used for transistor sizing in which the power dissipation is minimized while keeping the circuit delay unchanged. The circuit delay and power dissipation of initial circuits and optimized circuits are evaluated by a transistor-level static timing analysis tool[11] and a transistor-level power simulator[12], respectively. Fig. 6 show the results of power optimization for the circuits with minimum delay. On average, power reduction of 55 % is achieved without increasing circuit delay. For the circuits with minimum area, power reduction of 49 % is observed. Due to the reduction in power dissipation, peak current in the circuit is also reduced. In des circuit, for example, the peak current is decreased by 66 % which alleviates IR drop problem and enhances electro-migration reliability.

Similar to the power optimization, the amount of crosstalk noise voltage can also be reduced by transistor sizing. The transition rate of an aggressor signal can be reduced by downsizing the aggressor gate. A preliminary experiment on des and dsp_alu reveals that about 35 % reduction in maximum noise voltage is achieved without sacrificing speed performance of the circuits[13].

Conclusion

This paper describes a design methodology of ASICs that utilizes on-demand generation of libraries tailored to each application. The objective of the methodology is to obtain full custom performance in a cell-based design environment. According to the performance estimation of the circuit under design, a library that is best suited to the application is generated. A symbolic layout method is adopted to generate standard cells with variable driving strength. Due to the variability, a library with a rich set of driving strength can be supplied to cell-based design tools. After the completion of detailed routing, we can further optimize the circuit exploiting the tunability of driving strength. Design experiments as well as the measured result of a fabricated chip verify that the proposed design methodology gives better area-delay tradeoffs than a conventional cell-based design with a fixed library.

Acknowledgments

The authors thank A. Hirata, D. Fukuda, M. Takahashi, K. Fujimori, and H. Kanbara for their contribution to the development of on-demand library generation system. This work is supported in part by Semiconductor Technology Academic Research Center (STARC).

References

[1] D. G. Chinnery and K. Keutzer, ''Closing the gap between ASIC and custom: An ASIC perspective,'' in *Proc. of 37th DAC*, pp. 637 -- 642, June 2000.

[2] J. L. Burns and J. A. Feldman, ''C5M --- A control-logic layout synthesis system for high-performance microprocessors,'' *IEEE Trans. CAD*, Vol.17, No.1, pp. 14--23, Jan. 1998.

[3] S. Gavrilov, et. al., ''Library-less synthesis for static CMOS combinational logic circuits,'' in *Proc. of ICCAD'97*, pp. 658 --662, Nov. 1997.

[4] Tetsutaro Hashimoto and Hidetoshi Onodera, ''Layout generation of primitive cells with variable driving strength,'' in *Proc. of SASIMI 2000*, pp. 122-- 129, Apr. 2000.

[5] M. Guruswamy, et. al. , ''CELLERITY: A fully automatic layout synthesis system for standard cell libraries,'' in *Proc. of DAC'97*, pp. 327 -- 332, June 1997.

[6] Binary Ackalloor and Dinesh Gaitonde, ''An overview of library characterization in semi-custom design,'' in *Proc. of CICC'98*, pp. 305 -- 312, May 1998.

[7] A. Hirata, H. Onodera, and K. Tamaru, ''Proposal of a timing model for CMOS logic gates driving a CRC π load,'' in *Proc. on ICCAD'98*, pp. 537--544, Nov. 1998.

[8] H. Kurisu, ''I developed a CPU in this way!'' Design Wave Magazine, CQ Publishing company, pp. 35--82, Nov. 1999.

[9] T. Iwahashi, et. al., ''Vector Quantization Processor for Mobile Video Communication,'' in *Proc. of ASIC/SOC Conf. 2000*, pp. 75--79, Sept. 2000.

[10] Masanori Hashimoto and Hidetoshi Onodera, ''Post-layout transistor sizing for power reduction in cell-based design,'' in *Proc. of ASP-DAC 2001*, to appear, Jan. 2001.

[11] PathMill Reference Manual. Synopsys, Inc., CA, 1999.

[12] PowerMill Reference Manual. Synopsys, Inc., CA, 1999.

[13] Masanori Hashimoto, ''A study on performance optimization for digital CMOS circuits in physical design,'' PhD. dissertation, Kyoto University, Feb. 2001.