Process-Portable and Programmable Layout Generation of Digital Circuits in Advanced DRAM Technologies

Youngbog Yoon
DRAM Design Team
SK Hynix
Icheon, Korea
youngbog.yoon@sk.com

Daeyong Han
DRAM Design Team
SK Hynix
Icheon, Korea
daeyoung.han@sk.com

Shinho Chu
DRAM Design Team
SK Hynix
Icheon, Korea
shinho.chu@sk.com

Sangho Lee
DRAM Design Team
SK Hynix
Icheon, Korea
sangho2.lee@sk.com

Jaeduk Han
Electronic Engineering
Hanyang University
Seoul, Korea
jdhan@hanyang.ac.kr

Junhyun Chun
DRAM Design Team
SK Hynix
Icheon, Korea
junhyun.chun@sk.com

***Abstract*—This paper introduces a physical layout design methodology that produces DRC-clean, area-efficient, and programmable layouts of digital circuits in advanced DRAM processes. The proposed methodology automates the layout generation process to enhance design productivity, while still providing rich customization for efficient area and routing resource utilizations. Process-specific parameterized cells (PCells) are combined with process-independent place-and-route functions to automatically generate area-efficient and programmable layouts. Routing grids are optimized to enhance the area and routing efficiency. The proposed method reduced the design time of digital layouts by 80% compared to a manual design with high layout qualities, significantly enhancing the design productivity.**

Keywords—DRAM, Standard cells, Layout, Design automation, Templates

I. Introduction

The advance of semiconductor technology has enabled large-scale integrations of transistor devices on silicon wafers [1]. The dimension of active devices has been reduced to below 7-nm in channel length due to the use of 3-D structured devices (such as FinFET and FDSOI) and multiple-patterning technologies. However, these device-level innovations for process scaling often complicate device and interconnect structures [1], increasing design complexity. In order to minimize overheads in design productivity due to the complex device structures and design rules, cell-based design methodologies [2] have been widely adopted to digital circuits design in CMOS logic processes by properly abstracting design rules and promoting reuse of primitive cells. Although various digital circuits have been used in DRAM systems as well (Fig. 1), the cell-based approach has been discouraged in DRAM processes due to their stringent area and routing constraints. Physical layouts of digital cells in DRAMs must be heavily customized and reconfigured to environmental conditions (neighboring cells, routing patterns), thus limiting the reuse of identical logic cells across functional blocks. Instead, primitive digital cell layouts in DRAM processes are manually (re)designed for each use case, significantly increasing design efforts and cost.

In order to address the productivity issue in DRAM processes, this paper proposes an automated and efficient standard cell layout generation methodology for advanced DRAM technologies. The proposed method combines parameterized cells (PCells) with Python-based place-and-route functions to construct digital circuits with rich parameterization and customization capabilities. The place-and-route steps executed by layout generators are process-portable to enhance the design productivity and flexibility, with additional grid optimizations for higher area utilizations.

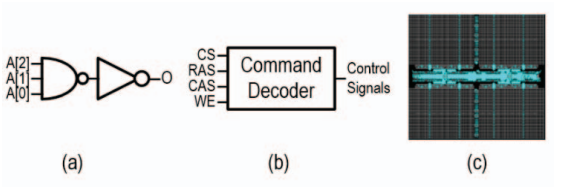

Fig. 1. Digital circuits in DRAM technologies (a) address decoder (b) control logic (c) layout of digital circuits in a DRAM chip.

II. Proposed Layout Generation Methodology of Digital Circuits

A. Design Considerations – Area and routing constraints

The area assigned to digital circuits in DRAM systems (for various control and data processing purposes) is basically 'wasted' in terms of storage capacity because the digital circuits cannot overlap with core memory cells. DRAM processes also have very limited routing layers (4-5 metal layers) compared to logic processes (usually have 8-12 metal layers), as the performance of DRAM is largely determined by their storage capabilities rather than routing flexibilities. As a result, digital circuits in DRAM chips adopt various customization techniques in device structures and routing patterns to reduce area and routing resource consumptions. Especially, poly and local interconnect (M0) patterns need to be widely reconfigured across designs, as illustrated in Fig. 2.

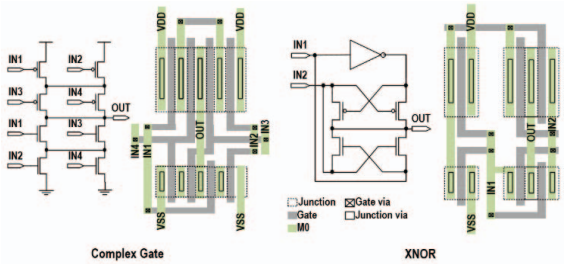

Fig. 2. Examples of complex digital cell layouts with various layout customizations in device placement and poly/M0 routings.

B. *Proposed layout generation framework with cell-level customizations and grid optimizations*

In order to implement digital circuit layouts with various customization options and meet the aforementioned requirements in DRAM processes, we developed an automated layout generation framework that produces optimal layouts to target technologies and design configurations, which is shown in Fig. 3. The automation and design optimization are achieved by combining the generator-based design methodology [3] for process-portability with technology-specific parameterized cells (PCells, [4]) for optimal layout structures and micro-customization capabilities. Instead of utilizing the generic and monotonic templates from [3], the proposed framework produces area-optimized core structures by executing Cadence SKILL-based PCell code with physical parameters and design parameters (such as pin locations and gate extensions). The process-specific parameters are abstracted in user levels as in [3], to make the generator code process-independent. This method achieves resource optimization and process portability, as long as PCells for core layout elements (such as stacked CMOS structures) are available (which is generally true because in-house PCells are widely used for conventional, manual design flows).

Another key aspect of the proposed approach is (re)defining the placement grid parameters with the generated and placed primitive elements, which is performed in the grid extraction and abstraction stages in Fig. 3. Instead of predefining fixed routing grids before executing generator code (which is utilized in [3]), the grid parameters are dynamically determined after the device placements, as illustrated in Fig. 4. The grid coordinates are still abstracted to support process portability and easy reconfigurations. The proposed approach yields extremely optimal utilization of routing tracks in DRAM processes.

III. Experimental Results

In order to verify the effectiveness of the proposed automatic generation flow, more than 300 logic circuits are generated in a 1b DRAM process, and their design qualities are evaluated. All pre-existing manual layouts are ported automatically to the new technology node without DRC errors, significantly enhancing the design productivity. While typically ~20 weeks/man has been assigned to produce the logic circuits in traditional design flows, it takes about 4 weeks/man to develop the generator code and generate the logic circuits. It is expected that the next development cycle should be shorter than 4 weeks/man, as most of the generator code can be reused. To compare the area consumptions of the manual and auto-generated layouts, their normalized areas are computed by the number of vertical routing tracks occupied by each cell. As shown in Fig. 5 (a), it turns out the generated designs consume almost the same vertical routing tracks (which is proportional to the layout area) compared to the pre-existing manual designs, which reveals the effectiveness of the highly customizable layout generation framework.

In terms of design productivity, the proposed method reduced the design time by more than 80% (including the generator code development time), as a significant portion of generator code is shared across designs. The layout generation framework is also utilized to generate various mixed-signal circuits (Fig. 5 (b)).

IV. Conclusions

An automatic layout generation framework is developed to produce optimal digital circuit layouts in advanced DRAM technologies, which directly impacts the design productivity in DRAM processes.

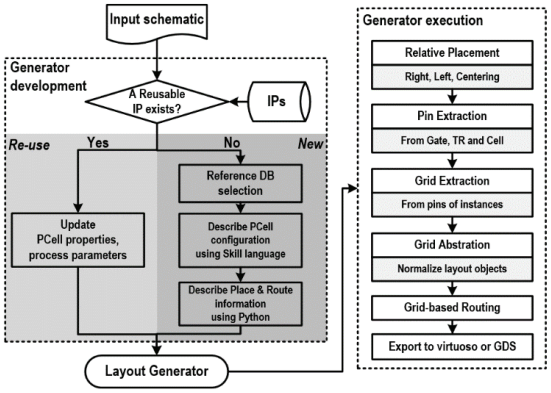

Fig. 3. Proposed digital circuit layout generation process.

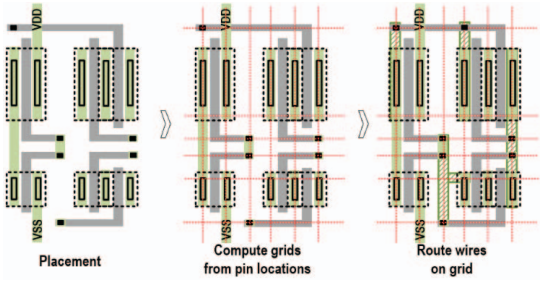

Fig. 4. Grid optimization process.

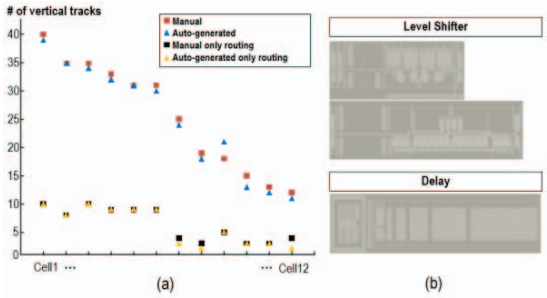

Fig. 5. (a) Design quality comparisons between manual and generated designs. (b) Example of generated mixed-signal circuits (blurred).

Acknowledgment

The authors extend special thanks to design transformation team members in SK Hynix for their valuable support.

References

[1] A. Loke *et al.*, "Analog/Mixed-Signal Design Challenges in 7-nm CMOS and Beyond," *Custom Integrated Circuits Conf. (CICC)*, 2018.

[2] B. Ren *et al.*, "A domain-specific cell based ASIC design methodology for digital signal processing applications," *Proceedings Design, Automation and Test in Europe Conference and Exhibition*, 2004.

[3] E. Chang *et al.*, "BAG2: A process-portable framework for generator-based AMS circuit design," *IEEE Custom Integrated Circuits Conference (CICC)*, 2018.

[4] H. Mathias *et al.*, "Automatic layout generation for CMOS analog transistors," *Proceedings of the conference on European design automation (EURO-DAC '94)*, 1994