A design method for process design kit based on an SMIC 65 nm process*

Luo Haiyan(罗海燕), Chen Lan(陈岚)†, and Yin Minghui(尹明会)

(Institute of Microelectronics, Chinese Academy of Sciences, Beijing 100029, China)

Abstract: The frame structure of a process design kit (PDK) is described in detail, and a practical design method for PDK is presented. Based on this method, a useful SMIC 65 nm PDK has been successfully designed and realized, which is applicable to native EDA software of Zeni. The design process and difficulties of PDK are introduced by developing and analyzing these parameterized cell (Pcell) devices (MOS, resistor, etc.). A structured design method was proposed to implement Pcell, which makes thousands upon thousands of source codes of Pcell concise, readable, easy-to-upkeep and transplantable. Moreover, a Pcase library for each Pcell is designed to verify the Pcell in batches. By this approach, the Pcell can be verified efficiently and the PDK will be more reliable and steady. In addition, the component description format parameters and layouts of the Pcell are optimized by adding flexibility and improving performance, which benefits analog and custom IC designers to satisfy the demand of design. Finally, the SMIC 65 nm PDK was applied to IC design. The results indicate that the SMIC 65 nm PDK is competent to support IC design.

Key words: PDK; Pcell; CDF
DOI: 10.1088/1674-4926/31/10/105008 **EEACC:** 2550X; 2560B; 2570A

1. Introduction

With the development of the technology for design automation and the demand for shorter design cycle times, the design method for analog/RF circuits faces a big challenge[1]. As a bridge to connect IC design and fabrication process, process design kit (PDK) plays a very important role in analog/RF design. PDK is a comprehensive set of foundry-verified data files, including parameterized cells (Pcells) used in an analog and mixed-signal/RF design flow[2, 3]. As the development of processes and the innovation of an EDA toolset, PDK benefits analog and custom IC designers by improving design efficiency, decreasing time-to-market (TTM)[1] and ensuring quality. The EDA toolset can not work effectively without process information and an external process design kit. There is a need to create a correlative PDK and to establish the design flow reflecting its design methodology. However, this is little considered during the development of a native EDA toolset, so neither the effective methods nor valuable experience is accumulated. Today, multiple foreign EDA vendors have their own way of developing PDKs. PDKs are written in proprietary languages and operated on proprietary databases. For example, Cadence has all-sided PDKs and Synopsys' EDK (open educational design kit)[4–7], Mentor's TDK and Agilent's Agilent DK have taken some of the market. But we have none of our own IP for PDK design used in our native software. In order to promote our technique, we propose a practical design methodology for native EDA software Zeni[8] with the example of a 65 nm PDK design. The design process and difficulties are introduced in detail. A comprehensive design methodology has been established. We have also proposed some efficient design techniques and reliable authentication methods to optimize the PDK. The 65 nm PDK which has been applied to real use has been validated, and can well support any analog and mixed-signal/RF circuits design. According to this project, we wish to be able to accumulate the practical design methods of a native EDA toolset so as to promote the use of our native software.

2. Design process and difficulties

As the element that interfaces the EDA toolset with the fabrication process, the construction of the PDK fills a critical role in the design process[2]. PDKs include such things as schematic symbols, Spice models, parameterized cells, a layout technology file, a design rule check (DRC) and layout-versus-schematic (LVS) deck, a parasitic extraction deck, a standard cell/pad library and associated characterization files and a framework to perform statistical circuit analysis[7].

A PDK is created from the fabrication process rules/data and a library of PDK templates at the EDA software of Zeni, as shown in Fig. 1. The fabrication process information is the application model of the PDK, as it represents the PDK abstracted from any toolset requirements. Different editors indicate which part of the PDK is to be generated. The translation engine indicates which PDK files will be generated from which source files and forms the customized PDK files. The generated PDK will then be applied to physical verification with real IC design.

Pcells are indispensable in PDK as it can not only ease layout by encapsulating the complex design rules but also allow the construction of the Pcell instance with any legal set of parameters which makes analog/RF circuit design more convenient[2]. But the construction of a Pcell is as complex as the set of design rules and device parameters[2].There is no unified scripting language for Pcell development: it can use languages such as Skill[9], TCL[10, 11] and Python. In order to ensure interoperability among all EDA vendor tools, and the reliability and expansibility of the code, we choose TCL scripting language to develop our Pcells. No matter what language, all the questions focus on how to design a strong, efficient, flex-

* Project supported by the National Major Specialized Program of China (No. 2008ZX01035-001-08).
† Corresponding author. Email: chenlan@ime.ac.cn
Received 14 April 2010, revised manuscript received 24 May 2010

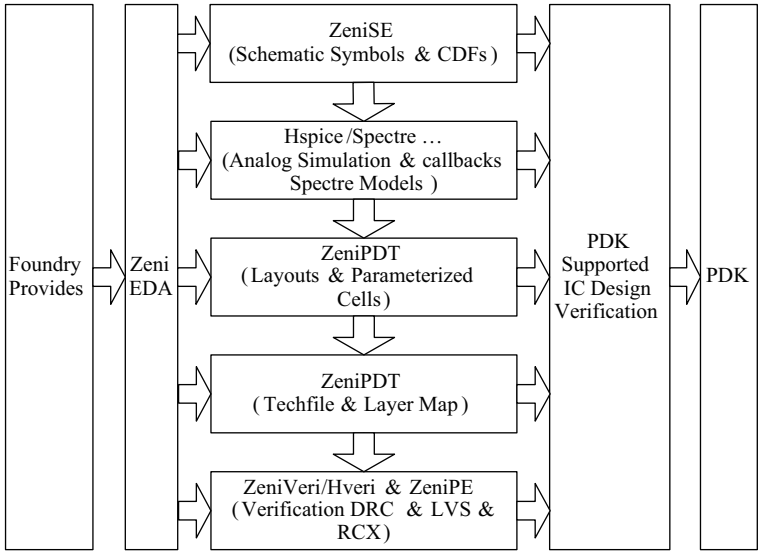

Fig. 1. Design procedure of PDK.

ible, readable and maintainable program to realize the control of devices and satisfy particular demands. A layout cell must be compiled from code, updated in the library and instantiated in another test layout to ensure that a piece of code is producing the desired result. Consequently, developing Pcells is a labor intensive coding task requiring greater effort and more design time compared with the manual layout of a fixed circuit[12].

For the sake of a successful PDK, the program must satisfy the conditions below: robustness; efficiency; flexibility; readability; maintenance. Thus, the PDK is competitive and superior.

3. Key parts and innovation

Pcells are the key parts of the PDK that use the structural specifications for the exact device that is (or the set of devices that are) needed in the layout, and produces such an instance to be placed in the layout design[2]. The Pcells are parameterized and their layouts must be described in a generalized way. This generalization results in more design time compared with a manual layout of a fixed circuit[10]. However, Pcells are favorable in design iterations as the regeneration of the layout is fairly fast. Further, Pcells simplify the reuse of the design since they can be modified to fit different specifications by changing parameter values, and the GUI of CDF (component description format) benefits analog and custom IC designers by improving efficiency. For example, during the analog/RF circuit design, if the width or the length of a transistor in an array is changed, a call-back function connected with the Pcell will translate the requested resistor width or length into resistance, thus making the design more efficient[2].

3.1. Parameterized cell (Pcell) realization

A Pcell is defined as a programmable cell that takes a set of input parameters to generate the layout of a cell[13]. We propose a structured design approach to developing Pcells. The rule of the structured approach is clear, readable and easy-to-update. By our approach, the Pcells not only have significantly fewer lines of codes, but also provide higher performance compared to traditional ones. The design flow of Pcells is shown in Fig. 2.

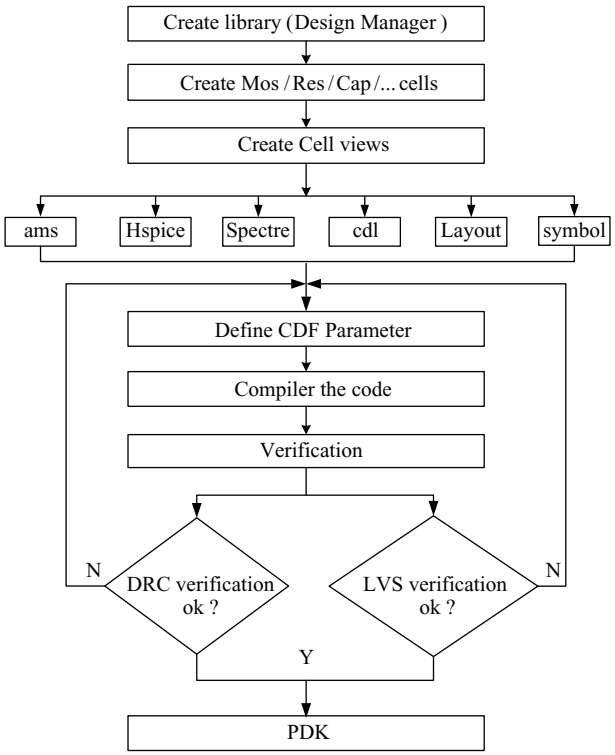

Fig. 2. Design flow of Pcell.

First, we need to create a library in which there are many device cells. Each cell has many cell views. Every cell view has its own CDF parameters and related function. For the Pcell, the CDF parameters and related function must be designed and realized correctly. We write the common function of the same kind of devices in terms of subsets. Usually they have many functions in common use. The source code of a certain Pcell can easily call the subset through passing correlative parameters. In this way, we have realized the reuse of the functional

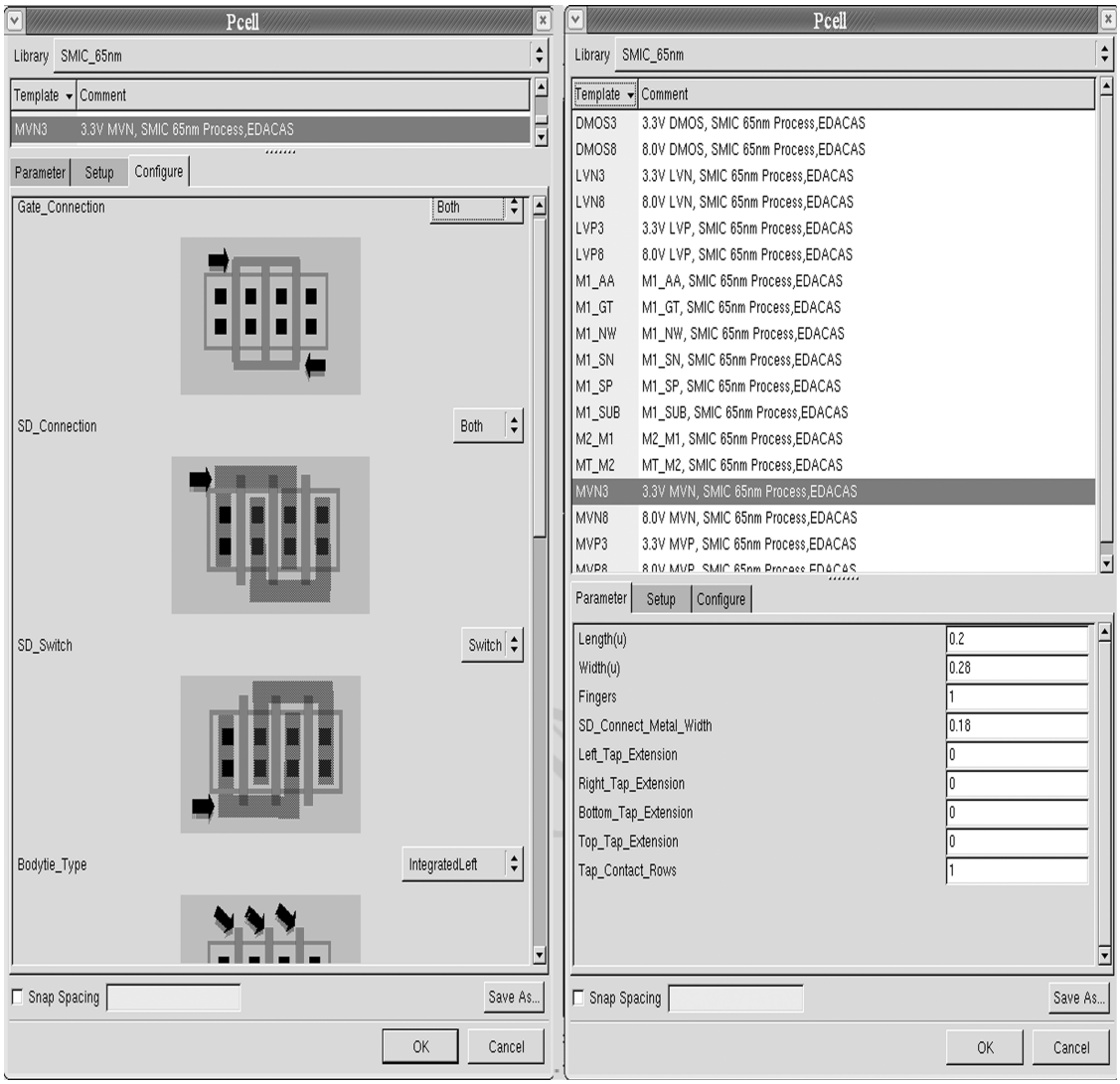

Fig. 3. CDF parameters of MOS.

modules, satisfying the various kinds of manufacture process and diversiform devices. Further, a layout cell will be compiled from code, updated in the library and instantiated in (or refreshed within) another test layout to ensure that the piece of code has produced the desired result. Afterwards, the layout modules must pass the DRC, which is executed in batches, and LVS verification before it becomes available. Therefore, there comes a successful and practical PDK. The Pcell is very flexible: the designer can easily call it to generate the layout of a cell according a set of real-time input parameters. Pcells can also realize the layout of more complex structures with the TCL program. An IC designer can visit these basic cells to realize his design conveniently and efficiently. The graphic interface achieved is shown in Figs. 3 and 4.

Taking the MOS-transistor and resistant as an example, we can see that the structured design approach has realized the CDF parameter perfectly: it has optimized the source code, made the design process more flexible and largely decreased the design time.

3.2. Innovation

Compared to conventional Pcells, we have optimized the CDF parameters and layouts of the Pcells from the circuit designer's view to satisfy the demand of design, which adds flexibility and convenience to IC designers.

Snake connection of the MOS makes the design more free as shown in Fig. 5.

The metal between the gate and drain become corrigible, making the layout more flexible.

Pcells support advance features, such as the width of guardring, which is revisable, and the rows of contacts on it can be modified.

The PDK based on Pcells that we optimized above will improve design productivity and ultimately promote innovation in analog and full custom EDA.

4. Physical verification

The design rule check (DRC) and layout versus schematic (LVS) become more and more important during the design. They play a significant role in eliminating errors, decreasing cost and improving productivity. Only when the layout has passed DRC and LVS verification can it be tapped out[2].

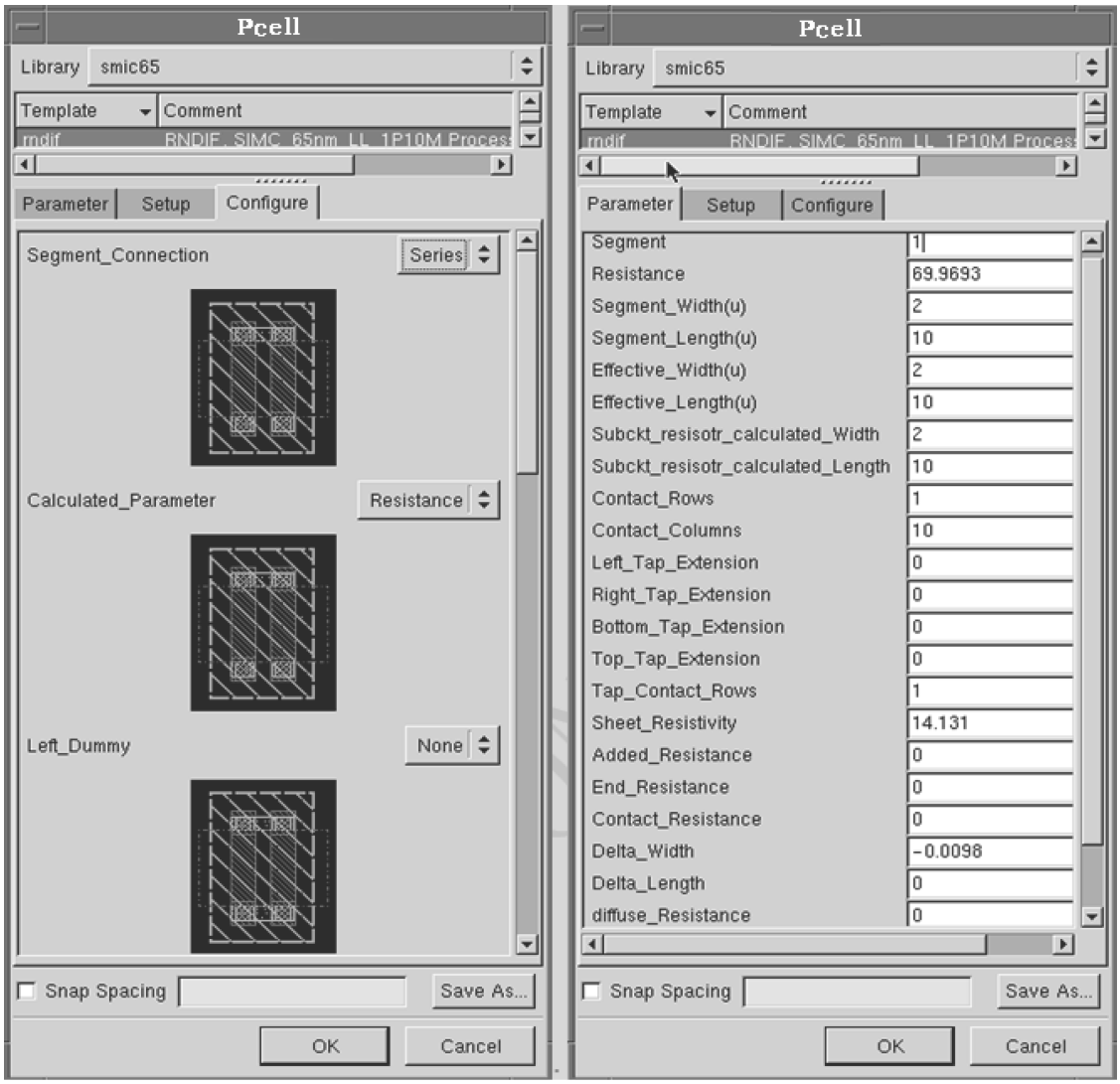

Fig. 4. CDF parameters of Res.

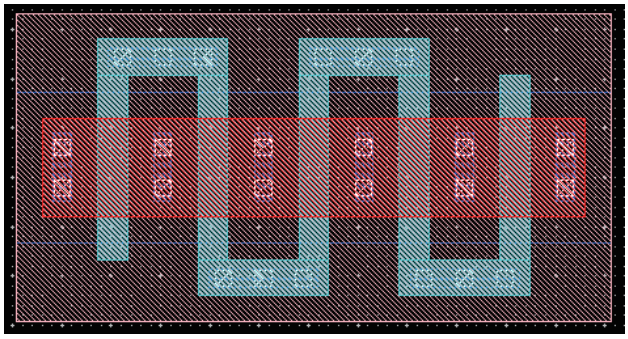

Fig. 5. Snake connection of the MOS.

4.1. Verify the Pcell's DRC in batches

A Pcase library for each kind of Pcell is designed to verify the Pcell with different combinations of parameters in batches, as shown in Fig. 6. We write test vector sets for each device to ensure their validity. Every test set has thousands of layouts, covering almost all the combinations of Pcell parameters and all the DRC rules. By this approach, we can verify Pcells efficiently and ensure that all of the layouts satisfy the design rules.

The test set supplies a whole and high coverage test vector, which can exactly test all the forms of the parameters of the Pcell. This method can not only ensure the quality of the Pcells but can largely improve the efficiency that manual verification can not achieve.

4.2. PDK validation

In this section, we discuss the validation of the whole SMIC 65 nm PDK created using TCL to ensure the validity of our Pcells. The flow of PDK validation is shown in Fig. 7. A 0.9 V bandgap circuit will be designed with the 65 nm PDK with the EDA software of Zeni[8] from schematic design to post-simulation with the parasitic parameters as shown in Fig. 8. The simulation result must be compared to the result of the real test to validate the PDK.

After DRC and LVS, we got the post-simulation results through XRC, and confirmed the reliability of our PDK by comparing with the pre-simulation results in Fig. 9.

The range of the output voltages is the same, though the waveform of the post-simulation drops a little behind that of

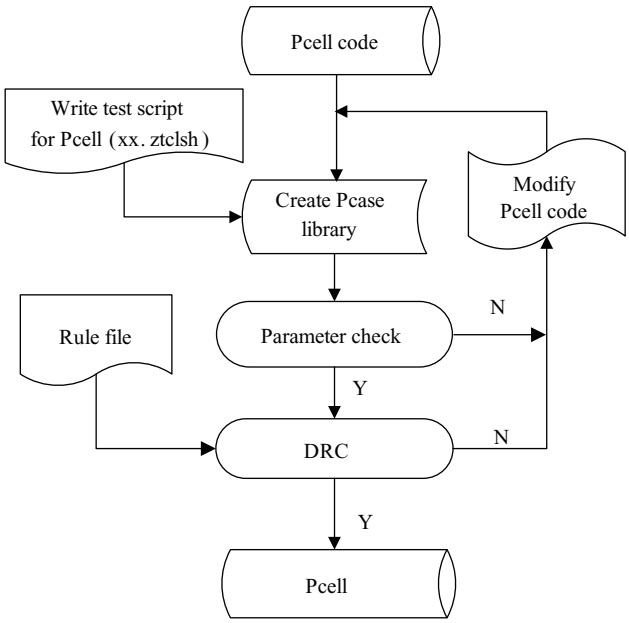

Fig. 6. Flow of DRC verification.

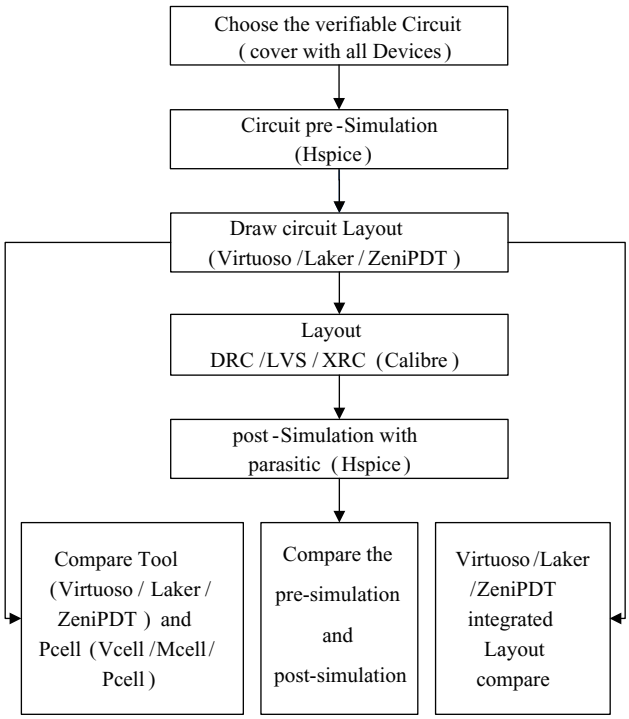

Fig. 7. Flow of PDK validation.

the pre-simulation. The function of the chip is to test that it is correct after it has been tapped out. The contrast showed high reliability and efficiency of our PDK, showing that it is competent to support IC design.

5. Conclusion

We have firstly presented a practical design method for PDK with the native EDA software of Zeni. The design of critical parts is described in detail and the application of 65 nm PDK in real circuit design shows high reliability and efficiency.

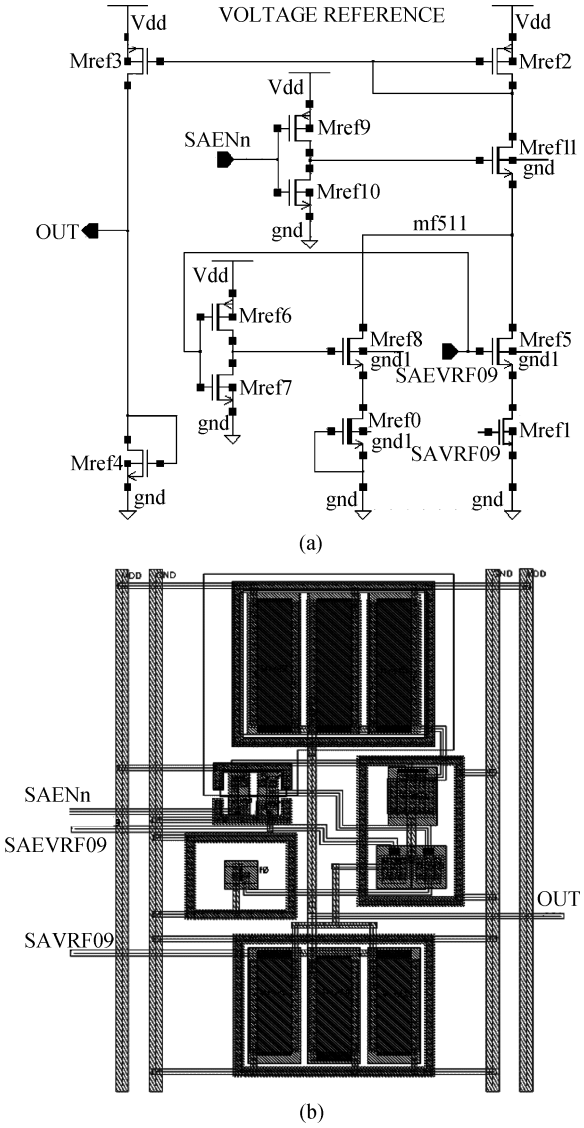

Fig. 8. (a) Circuit and (b) layout of the 0.9 V bandgap.

The structured design method we suggest has made thousands upon thousands of source codes of Pcell concise, readable, easy-to-upkeep and transplantable. Moreover, we optimized the CDF parameters and layouts of the Pcells to satisfy the demand of the design, which has made the designs more flexible and convenient. In addition, compared to the conventional approach, we have designed a Pcase library for each Pcell to verify the Pcells in batches. Based on this method, we can verify Pcells efficiently and ensure that the PDK is more reliable and steady.

With the successful design and realization of a SMIC 65 nm PDK, we achieve concise source codes, verified Pcells, optimized layouts, etc. This is a significant advance in PDK development. The design approach of the PDK and the verification method we propose will enable design reuse and improve design productivity. The designers are able to settle out of the complex and difficult work and become more convenient and efficient. Moreover, it is a big contribution to the innovation of native IP design, which is valuable and attractive in the competitive world today.

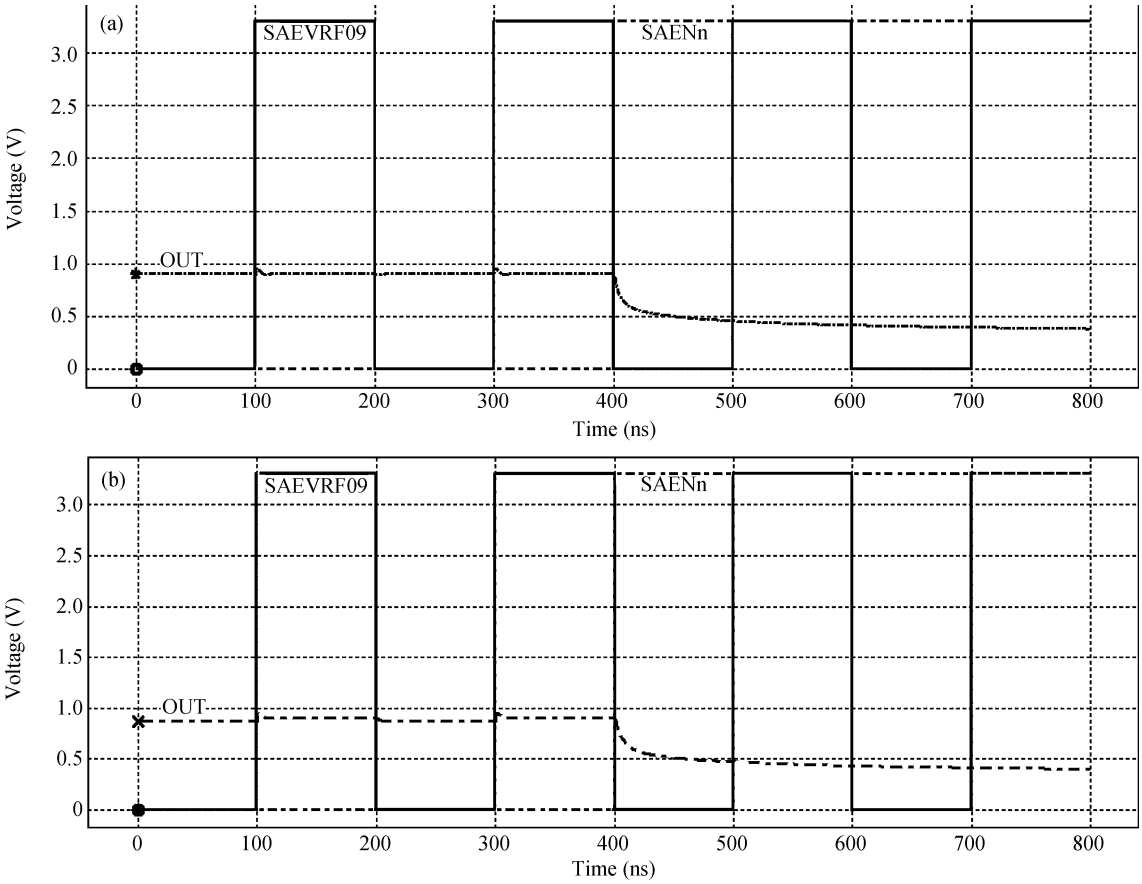

Fig. 9. Contrast of (a) pre-simulation and (b) post-simulation of a 0.9 V bandgap circuit.

References

[1] Gielen G G E. New methods and tools for analog IP use in mixed-signal integrated systems. Proc NORCHIP Conference, Kista, Sweden, Nov 12–13, 2001: 5

[2] Fishe J S. Application of model driven architecture design methodologies to mixed-signal system design projects. PhD Thesis, Ohio State University, 2006

[3] Chang C S, Chao C P, Chern J G J, et al. Advanced CMOS technology portfolio for RF IC applications. IEEE Trans Electron Devices, 2005, 52(7): 1324

[4] Goldman R, Bartleson K, Wood T. Synopsys' open educational design kit: capabilities, deployment and future. Synopsys Inc., CA, USA, Synopsys Armenia CJSC, Yerevan, Armenia

[5] Stine J E, Castellanos I, Wood M, et al. Free PDK: an open-source variation-aware design kit. IEEE International Conference on Microelectronic System Education, 2007: 173

[6] Stine J E, Chen J, Castellanos I, et al. Free PDK v2.0: transitioning VLSI Education towards nanometer variation-aware designs. IEEE International Conference on Microelectronic Systems Education, 2009:100

[7] Basavarajaiah S. Circuit extraction and simulation in the presence of random and systematic process variations. MS Thesis, North Carolina State University, 2008

[8] Zeni Schematic/Symbol Editor User Guide, Copyright ©_ 2002-2008, HED

[9] Virtuoso parameterized cell reference, product Version 5.0. Cadence Design Systems Inc., USA, July 2002

[10] Todorovich E, Cadenas O. TCL/TK for EDA tools. IEEE 3rd Southern Conference on Programmable Logic, 2007: 107

[11] Ousterhout J K. TCL and the TK toolkit. Addison-Wesley Reading, Mass, 1994

[12] Andersson K O, Vesterbacka M. A parameterized cell-based design approach for digital-to-analog converters. Proceedings of the 4th IEEE International Workshop on System-on-Chip for Real-Time Applications, 2004: 225

[13] Uehara T, Vancleemput W M. Optimal layout of CMOS functional arrays. Proceedings of the ACM/IEEE Design Automation Conference, 1978: 287