iEDA: An Open-Source Intelligent Physical Implementation Toolkit and Library

Xingquan Li[1,11*], Simin Tao[1], Zengrong Huang[1], Shijian Chen[1,2,13], Zhisheng Zeng[1,2,13], Liwei Ni[1,2,13], Zhipeng Huang[1], Chunan Zhuang[1], Hongxi Wu[4,3], Weiguo Li[11,3], Xueyan Zhao[2,13,1], He Liu[5,1], Shuaiying Long[1], Wei He[1], Bojun Liu[1], Sifeng Gan[1], Zihao Yu[2,3], Tong Liu[12,3], Yuchi Miao[1], Zhiyuan Yan[2,13], Hao Wang[6,3], Jie Zhao[1], Yifan Li[7,13,3], Ruizhi Liu[2,13,1], Xiaoze Lin[1,2,13], Bo Yang[1,9], Zhen Xue[1,2,13], Zonglin Yang[8,3], Zhenggang Wu[1], Jiangkao Li[11,3], Yuezuo Liu[6,3], Ming Peng[6,3], Yihang Qiu[10,3], Wenrui Wu[8,3], Zheqing Shao[6,3], Kai Mo[8,3], Jikang Liu[8,3], Yuyao Liang[8,3], Mingzhe Zhang[6,3], Zhuang Ma[6,3], Xiang Cong[6], Daxiang Huang[1], Guojie Luo[5,1], Huawei Li[2,13,1], Haihua Shen[13,1], Mingyu Chen[2,13,1], Dongbo Bu[14,13,1], Wenxing Zhu[4,1], Ye Cai[8,1], Xiaoming Xiong[10], Ying Jiang[9,1], Yi Heng[9,1], Biwei Xie[2,1,✉], and Yungang Bao[2,13,3,1,✉]

[1]Peng Cheng Laboratory, Shenzhen, China
[2]State Key Lab of Processors, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
[3]Beijing Institute of Open Source Chip, Beijing, China
[4]Fuzhou University, Fuzhou, China
[5]Peking University, Beijing, China
[6]University of Science and Technology of China, Hefei, China
[7]Institute of Microelectronics, Chinese Academy of Sciences, Beijing, China
[8]Shenzhen University, Shenzhen, China
[9]Sun Yat-sen University, Guangzhou, China
[10]Guangdong University of Technology, Guangzhou, China
[11]Minnan Normal University, Zhangzhou, China
[12]The Hong Kong University of Science and Technology (Guangzhou), Guangzhou, China
[13]University of Chinese Academy of Sciences, Beijing, China
[14]Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
Email: *lixq01@pcl.ac.cn, ✉xiebiwei@ict.ac.cn, ✉baoyg@ict.ac.cn

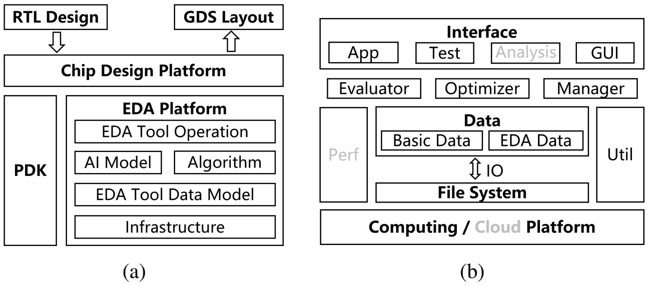
***Abstract*—Open-source EDA shows promising potential in unleashing EDA innovation and lowering the cost of chip design. This paper presents an open-source EDA project, iEDA, aiming for building a basic infrastructure for EDA technology evolution and closing the industrial-academic gap in the EDA area. iEDA now covers the whole flow of physical design (including Floorplan, Placement, CTS, Routing, Timing Optimization etc.), and part of the analysis tools (Static Timing Analysis and Power Analysis). To demonstrate the effectiveness of iEDA, we implement and tape out three chips of different scales (from 700k to 1.5M gates) on different process nodes (110nm and 28nm) with iEDA. iEDA is publicly available from the project home page http://ieda.oscc.cc.**

***Index Terms*—Netlist-to-GDS, infrastructure, tool, flow, chip.**

I. Introduction

The rapid advancement of digital technology has led to a substantial increase in demand for the growth of the integrated circuit (IC) industry. However, Moore's law has reached its limits [1]. To meet the growing needs for computing and storage, we may explore alternative integration technologies, such as 3D chip design or package and chiplets. Moreover, we should strive to improve the quality of chip design, especially in the area of electronic design automation (EDA). Chip design, EDA tools, and methodologies have been the subject of research for many years. Fortunately, new techniques such as artificial intelligence, hardware acceleration, and collaborative design of chips and EDA have demonstrated their capacity and practical value in EDA. These advancements open up the potential for further optimizing EDA tools and methodologies.

To maximize the benefits of new techniques in EDA, we need an open-source platform accessible to all EDA enthusiasts. There are already several open source physical design platform OpenROAD [2] and tools (e.g., DREAMPlace [3], CUGR [4], Dr.CU 2.0 [5], NTHU-Route [6], OpenTimer [7]) that work well to promote open collaboration and innovation. The platform should be automated, well-designed, and compatible while supporting tapeout while offering high performance and scalability, providing comprehensive documentation, and easy-to-learn. The iEDA project aims to create an open-source EDA platform and build an EDA community. It supports chip implementation from netlist to GDSII for block-level chips and system-on-chips (SoCs) at 28nm process node. The main goal of iEDA is to provide a research platform for enthusiasts to explore chip design, EDA methodologies, algorithms, software development, and more.



(a)

(b)

Fig. 1. The iEDA framework and infrastructure.

II. iEDA Status

A. Framework

The iEDA platform consists of four key components: basic infrastructure, a functional module-level data model, a solver (algorithms and AI models), and function operations as shown in Fig. 1(a). The infrastructure includes the basic data and a series of EDA software kits and frameworks. The data model provides pre-design data services for upper-level EDA functions. The AI models and algorithms are the core solvers that determine the quality and performance of the platform. The operations component is responsible for chip design functions, which include data models and solvers. The iEDA platform, along with a process design kit (PDK), enables the flow of chip design from RTL design to GDS layout.

B. Infrastructure

The iEDA software infrastructure, as shown in Fig. 1(b), includes a file system with parsers and file managers, a database with builders, writers, and data managers, a platform with evaluators, optimizers, and managers, and an interface with apps, evaluations, and tests, as well as a GUI. This unified infrastructure ensures the robustness and extensibility of iEDA.

C. Tools

The chip design step and the iEDA tools are shown in Fig. 2, each iEDA tool is composed of several low-coupling functional operations, which works by calling a series of different algorithms on designated data models.

1) Floorplan and Power Delivery Network: Floorplanning is the initial stage of physical design. iFP mainly includes the following parts: layout initialization, pre-placement of certain cells, automatic generation of IO locations, automatic placement of macros based on the netlist structure and IO, and generation of the power delivery network.

2) Placement: Placement mainly ensures the proper coordinate of each cell mapped in the netlist within a designated region, which must comply with design rules and be conducive to routing, timing convergence, and power consumption. iPL is mainly composed of a global placer, post-global placer, legalizer, detailed placer, filler, and checker. The objective of iPL is to minimize wire-length, timing and congestion.

3) Timing Optimization: Timing optimization is aimed at ensuring that the chip design is both functionally correct and meets the design requirements for performance. iTO offers distinct and user-friendly interfaces to fix timing design rule violations, hold time violations, and setup time violations. Our goal is to resolve any timing violations on the chip to enhance its overall performance.

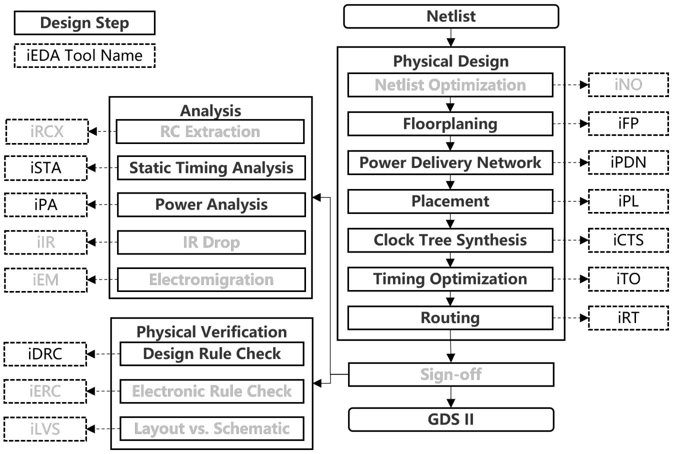


Fig. 2. Chip design step and iEDA tools.

4) Clock Tree Synthesis: The goal of iCTS is to balance skew among flip-flops while optimizing design resource usage, under timing constraints. It has an integrated timing model that offers accurate and efficient computation of timing information, such as delay, capacitance, and slew. The algorithms implemented in iCTS allow for the generation of topologies with various objectives and support the creation of hierarchical clock trees using multiple merging criteria.

5) Routing: Routing entails physical wires of the interconnections among components embedded in a chip. The objective of routing is to meet performance and functional requirements while considering factors such as design rule check (DRC) and signal integrity. iRT consists of components such as a resource allocator, topology generator, planar router, layer assigner, guide processor, track assigner, space router, and region manager.

6) Static Timing Analysis: iSTA is a timing analysis tool that provides easy-to-use interfaces for accessing required timing data. Besides of basic timing path analysis, iSTA offers incremental timing propagation, detailed timing reports, advanced delay calculation based on composite current source (CCS) liberty, basic cross-talk delay analysis, and on-chip variation analysis. iSTA also supports parallel processing of load liberty data for faster performance.

7) Power Analysis: iPA is a power analysis tool that supports both basic vectorless analysis without waveform data and vector analysis with VCD/SAIF data. It features fast toggle and static probability propagation, and glitch analysis that considers timing delay.

D. Flow

To verify the functionality and effectiveness of iEDA, we develop iFlow, which is implemented in Python by calling TCL-based or Python-based tool commands. It is worth noting that, since iEDA supports only netlist-to-GDSII, iFlow integrates Yosys/ABC [8] [9] for synthesis and some commercial tools for signoff analysis and verification. We implement and

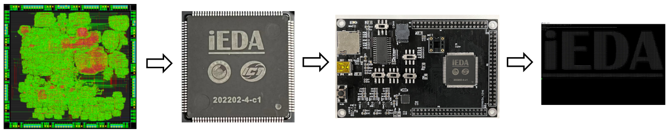


Fig. 3. From GDS to computer device.

tape out three chips with iEDA: one 5-stage RISC-V chip supporting RT-Thread on 110nm; two 11-stage RISC-V chips supporting Linux on 110nm and 28nm in respective. The first 5-stage RISC-V chip on 110nm has already successfully booted up (Fig. 3), and the other two chips will return before May.2023.

III. iEDA Future

A. Infrastructure

To better serve the research and development of EDA technology, we will further support more file formats and process nodes, and provide more evaluators and optimizers. In addition, we will try to decompose the tools of iEDA into micro-services and build a cloud-native EDA system.

B. Tools

The performance of physical design tools will be further improved through carefully designed software architecture and key algorithms, combined with hardware acceleration and AI models, to support larger and more diverse chip designs and scenarios. We will launch electromigration (EM) and interconnect resistance drop (IR) analysis to complete the analysis toolchain. All analysis tools will share the common database with physical design tools. Physical verification is the final step in IC design closure and typically involves design rule check (DRC), layout versus schematic (LVS), and electrical rule check (ERC). In the upcoming version of iEDA, the development of these tools is being planned.

C. Flow

In the era of intelligence, it has gradually become a trend to use artificial intelligence technology to liberate engineers from heavy repetitive work. This can also further improve the chip performance and production efficiency. To achieve this goal, we plan to extract most of the parameters in the design flow and optimize chip design objective by means of design space exploration.

Acknowledgment

This work is supported in part by the Major Key Project of PCL (No. PCL2021A08), the Strategic Priority Research Program of Chinese Academy of Sciences (No. XDC05030400), the Strategic Priority Research Program of Chinese Academy of Sciences (No. XDA0320300), the National Key R&D Program of China (No. 2022YFB4500403), National Natural Science Foundation of China (No. 62090024), the National Natural Science Foundation of China (No. 62090021), the National Natural Science Foundation of China (No. 62174033), the National Natural Science Foundation of China (No. 61907024).

References

[1] M. S. Lundstrom and M. A. Alam. Moore's law: The journey ahead. *Science*, 378(6621):722–723, 2022.

[2] T. Ajayi and D. Blaauw. OpenROAD : Toward a self-driving, open-source digital layout implementation tool chain. In *Proceedings of Government Microcircuit Applications and Critical Technology Conference*, 2019.

[3] Y. Lin, S. Dhar, W. Li, H. Ren, B. Khailany, and D. Z. Pan. DREAMPlace: Deep learning toolkit-enabled GPU acceleration for modern VLSI placement. In *Proceedings of the 56th Annual Design Automation Conference*, pages 1–6, 2019.

[4] J. Liu, C.-W. Pui, F. Wang, and E. F. Young. CUGR: Detailed-routability-driven 3D global routing with probabilistic resource model. In *Proceedings of the 57th ACM/IEEE Design Automation Conference*, pages 1–6. IEEE, 2020.

[5] H. Li, G. Chen, B. Jiang, J. Chen, and E. F. Young. Dr. CU 2.0: A scalable detailed routing framework with correct-by-construction design rule satisfaction. In *Proceedings of IEEE/ACM International Conference on Computer-Aided Design*, pages 1–7. IEEE, 2019.

[6] Y.-J. Chang, Y.-T. Lee, and T.-C. Wang. NTHU-Route 2.0: A fast and stable global router. In *Proceedings of IEEE/ACM International Conference on Computer-Aided Design*, pages 338–343. IEEE, 2008.

[7] T.-W. Huang and M. D. Wong. OpenTimer: A high-performance timing analysis tool. In *Proceedings of IEEE/ACM International Conference on Computer-Aided Design*, pages 895–902. IEEE, 2015.

[8] C. Wolf, J. Glaser, and J. Kepler. Yosys - A free verilog synthesis suite. In *Proceedings of the 21st Austrian Workshop on Microelectronics*, 2013.

[9] R. Brayton and A. Mishchenko. ABC: An academic industrial-strength verification tool. In *Proceedings of Computer Aided Verification, Edinburgh, UK*, pages 24–40. Springer, 2010.