Guest Editors' Introduction: The Resurgence of Open-Source EDA Technology

Sherief Reda
Brown University

Pierre-Emmanuel Gaillardon
University of Utah

Leon Stok
IBM

In the 1980s, the academic community produced several very high-quality electronic design automation (EDA) tools that spawned the EDA industry. Tools such as Spice [1], Espresso [2], and SIS [3] became the foundation of EDA companies. Open-source tools enable rapid innovation and create an ecosystem for scientific development. In recent years, the cost and difficulty involved in the design of integrated circuits (ICs) in advanced nodes have stifled hardware design innovation and have raised unprecedented barriers to bring new design ideas to the marketplace. Unlike the thriving software community, which enjoys a large number of open-source operating systems, compilers, libraries, and applications, the hardware community lacks such a modern ecosystem. With the advent of open silicon IP ecosystems, such as RISCV, Chips Alliance, and Free Silicon Foundation, the time has come to reinvigorate the open-source movement in EDA tools. Recent programs from governmental agencies aim to jump-start the development of open-source EDA tools to reduce the cost and turnaround time of hardware design.

Digital Object Identifier 10.1109/MDAT.2020.3038851
Date of current version: 8 April 2021.

The availability of open-source EDA tools leads to multiple benefits. First, the availability of open-source tools leads to reproducible research with clear identification of state-of-the-art results. Thus, open-source tools enable unequivocal benchmarking that can quickly identify new EDA solutions that advance the state of the art. Second, open-source tools enable the acceleration of EDA research as innovations can be implemented at a faster rate by building on top of existing open-source tools and components. Thus, open-source tools lower the barrier to entry to the field by new students or practitioners. Third, full-stack open-source tools enable the quantification of improvements across the entire EDA flow. Since it is possible that improvements in one EDA stage are masked by downstream tools, evaluation within the context of a full stack of open-source EDA tools ensures that these improvements stick till the end. Fourth, open-source tools with standard I/O format exchanges enable a healthy ecosystem to develop between open-source tools and closed-source industrial tools, leading to faster dissemination of knowledge between academia and industry. Fifth, open-source EDA tools lead to a more trustworthy design process since the scrutinizing of an open-source tool by a community of developers can identify any backdoors that lead to the capture

Copublished by the IEEE CEDA, IEEE CASS, IEEE SSCS, and TTTC

of sensitive design information or potential insertion of hardware Trojans.

Open-source development leads to special challenges. First, there is a need for common infrastructure tools, such as EDA databases that consolidate shared tasks, such as data structures for internal circuit representations and reading/writing of standard I/O EDA formats. Second, there is a need for open-source tools to fully interoperate with physical design kits (PDKs) and libraries. Existing open-source PDKs and libraries (e.g., FreePDK45nm) do not map to any real manufacturing flows. However, recent efforts by Google and Skywater to release a full manufacturing PDK in 130 nm provide a hopeful path [4], where other vendors might follow suit and open-source the PDKs and libraries of some of their mature technology nodes. Third, open-source tools need maintenance beyond their release, requiring long-term commitment and funding. Thus, developing and engaging a community of developers through collaborative platforms, e.g., Github, is essential for long-term success.

In this issue, we have collected papers that touch upon key parts of the design flow. A requirement in the review process was that the code is released as open-source, and all the tools released with the special issue are given in Table 1.

The first series of articles introduces complete flows, addressing a specific design category, namely analog, synchronous digital, and asynchronous digital. First, analog layout tools are a key part of any electronic design system. They are as essential to analog design as they are to digital design where they are used to design the cell libraries, memory cells, and all key analog components. The paper titled "ALIGN: A System for Automating Analog Layout" describes a correct by construction approach to synthesize electrically and designs compliant design. By taking advantage of layout hierarchies the authors are able to apply this to an interesting class of circuits. The second paper on analog design flows is titled "MAGICAL: An Open-Source Fully Automated Analog IC Layout System from Netlist to GDSII," where it presents MAGICAL, which is a fully automated analog IC layout system. MAGICAL takes a netlist and design rules as inputs and it produces the final GDS layout in a fully automated fashion. For asynchronous logic flows, the paper titled "An Open-Source EDA Flow for Asynchronous Logic" presents an open-source EDA flow for digital asynchronous circuits, capable of supporting many different families of asynchronous circuit families from logic synthesis all the way down to GDSII. Finally, the paper titled "Real Silicon Using Open-Source EDA" demonstrates that complete open-source tooling can be used to design industrial quality digital circuits. Using the OpenLane framework, based itself on the OpenROAD tool [5], the authors show a complete set of RISCV-based SoC.

Table 1. Open-source tools released for articles in this special issue.

Tool	Source
ALIGN	https://github.com/ALIGN-analoglayout/ALIGN-public
MAGICAL	https://github.com/magical-eda/MAGICAL
Asynchronous logic	https://github.com/asyncvlsi/
OpenLane	https://github.com/efabless/openlane
Fault	https://github.com/Cloud-V/Fault
PyMTL3 and PyH2	https://github.com/pymtl/pymtl3
OpenTimer	https://github.com/OpenTimer/OpenTimer
CATNAP-Sim	http://esrlab.ce.sharif.ir/download/CATNAP-Sim.gz.tar

In addition to complete flows, the second series of articles introduces specific point tools. Design for test (DFT) is an integral part of the design flow. An open-source DFT flow is therefore essential for any open-source solution. The paper titled "Fault: Open-Source EDA's Missing DFT Toolchain" describes an approach to fill in this missing piece. The paper titled "PyH2: Using PyMTL3 to Create Productive and Open-Source Hardware Testing Methodologies" proposes a new model testing and verification methodology, PyH2, using property-based random testing in Python. PyH2 leverages the whole Python ecosystem to build test benches and models. The paper "OpenTimer v2: A Parallel Incremental Timing Analysis Engine" introduces a high-quality open-source static timing analysis engine that is capable of parallel incremental timing and that provides an efficient API to facilitate the development of complex EDA tools. Finally, the paper titled "CATNAP-Sim: A Comprehensive Exploration and a Nonvolatile Processor Simulator for Energy Harvesting Systems" introduces an architecture exploration tool to study and understand the tradeoffs of future processor systems using

nonvolatile memory and help guide the design of the future.

We hope you enjoy the articles and tools that are available with this special issue. ■

References

[1] L. W. Nagel and D. O. Pederson, "SPICE (Simulation Program with Integrated Circuit Emphasis)," EECS Dept., Univ. California, Berkeley, Tech. Rep. No. UCB/ERL M382, 1973.

[2] R. K. Brayton et al., *Logic Minimization Algorithms for VLSI Synthesis* (The Kluwer International Series in Engineering and Computer Science 2). New York, NY, USA: Springer, 1984, ISBN 978-1-4612-9784-0, pp. 1–193.

[3] E. M. Sentovich et al., "SIS: A system for sequential circuit synthesis," EECS Dept., Univ. California, Berkeley, Tech. Rep. No. UCB/ERL M92/41, May 1992.

[4] Google Skywater PDK. [Online]. Available: https://github.com/google/skywater-pdk

[5] T. Ajya et al., "Toward an open-source digital flow: First learnings from the OpenROAD project," in *Proc. ACM/IEEE Design Autom. Conf.*, 2019, Article no. 76, pp. 1–4.

Sherief Reda is a Full Professor with the School of Engineering, Brown University, Providence, RI, USA. His research interests are in the area of hardware systems, with focus on energy-efficient computing, design automation of integrated circuits, embedded systems, and computer architecture. He is a Senior Member of IEEE.

Leon Stok is Vice President of EDA IBM. Stok has a PhD degree in electrical engineering from the Eindhoven University of Technology, Eindhoven, The Netherlands. He is a Fellow of IEEE.

Pierre-Emmanuel Gaillardon is an Associate Professor with the Electrical and Computer Engineering (ECE) Department, The University of Utah, Salt Lake City, UT, where he leads the Laboratory for NanoIntegrated Systems (LNIS). He is a Senior Member of IEEE.

■ Direct questions and comments about this article to Sherief Reda, School of Engineering, Brown University, Providence, RI 02912 USA; sherief_reda@brown.edu.