Keynote

Strange Loops in Design and Technology: 59th DAC Keynote Speech

Giovanni De Micheli
EPFL
1015 Lausanne, Switzerland

Editor's notes:
This keynote paper highlights the interaction between emerging technologies and software tools to enable the current evolution of electronic design automation systems.
—*Partha Pratim Pande, Washington State University, USA*

DESIGN, MANUFACTURING, AND APPLICATIONS of electronic systems have reached unprecedented levels of quality, being fueled by the advances in hardware support and acceleration of *artificial intelligence* (AI) and *machine-learning* (ML) algorithms, the electrification and extended capabilities of vehicles, and the ubiquitous presence of terminals for wireless communication, such as phones. These achievements have roots in two pillars: 1) the advancement of CMOS nano-electronic technology and 2) the improvement of performance and capabilities of *EDA* tools. The thesis of this article is that there is a strong interaction among these pillars that can be seen as two intertwined spirals feeding on each other and bringing progress to new heights.

The notion of progress as the repetition of spiral-like patterns was proposed by enlightenment philosopher G. Vico, as stated within the EDA evolution perspective by Sangiovanni-Vincentelli [1] in the 40th DAC keynote address. In a similar vein, Pulitzer-winner Hofstadter [2] wrote on the importance of the circularity of knowledge in arts and sciences. Through this process, the innovation leap upward is often made possible: indeed, a *strange loop*, which does not close because of an unresolved constraint, is a stimulus for progress. When considering the *design technology* (also known as EDA) and *fabrication technology* of ICs and systems, there is a circular interaction of opportunities and challenges. This article will elaborate on three examples of the spiral-like evolution of design and technology. Other similar examples can be constructed in the software domain, such as the use of AI for EDA. Nevertheless, this article concentrates on examples linked to hardware and technology. The objective of this review is to promote an

open, broad, and far-sighted attitude in research, development, and marketing of new ideas and technologies, by looking beside traditional paths and across application domains.

Emerging technologies

One burning question for technology researchers, developers, and users is whether CMOS is invincible, and if post-CMOS technologies will be always the *technologies of the future*. There are strong dents into CMOS supremacy, in particular, by the use of *carbon nanotubes* (CNTs) and 2-D electronic devices. The former has been used to realize a 16-bit RISC V microprocessor, with about 15,000 complementary transistors [3]. The latter relies on sandwiches (possibly monolayers—hence the name) of transition metal dichalcogenides (e.g., MoS_2 WSe_2) to realize computational elements, ranging from logic gates to a perceptron [4], [18]. Thus, both technologies have been shown to be technically viable and competitive. Nevertheless, despite these and other efforts, any emerging technology has to satisfy some requirements to stay afloat, that is, to provide a product-level alternative to silicon. Namely, a new technology: 1) needs to address a particular aspect of design that cannot be realized in CMOS; 2) needs to perform better, in terms of speed, cost, and/or energy consumption; and 3) requires good quality (i.e., production-level) EDA tools. Today, CNTs and 2-D electronic devices have shown promising performance and a strong advantage (over CMOS) for realizing 3-D ICs [12]. These technologies provide us with computational strata that can be layered on top of silicon because devices can be fabricated with a low-temperature fabrication process. Nevertheless, the lack of EDA tools has limited their application to a research niche.

Another fundamental question is why computation needs various paradigms and technologies. The reason is that our insatiable appetite for computing and increasingly tighter budgets on energy consumption led us to use *hardware accelerators* and *application-specific units* for some part of the computation. Often, bit-level manipulation is more efficient than word-level processing. The most common example is to accelerate computation via *FPGAs* [6]. Alternatively, the structure of the computation matches well with the layout of the circuits that realize it. The search for smart applications of post-CMOS technologies is driven by the fact that diverse technologies can be more efficient than standard CMOS in solving some parts of the computation, for example, by means of *in-memory computing*. In particular, the problem of matrix–vector multiplication can be solved natively on a memory array, by storing the matrix entries in the memory, and by using an analog or digital readout. Such memories can be *nonvolatile* and realized through *back-end-of-line* (BEOL) steps on top of other CMOS circuits, thus providing both area savings as well as shorter and faster interconnect [13]. Recently, optical components have shown high-performance matrix–vector multiplication and good interfacing means to CMOS. Both these architectural solutions have a major impact on the execution of ML algorithms and AI applications.

The circular exploration of emerging technologies and devices and their related design tools can be exemplified by the following experience. About a decade ago, it became clear that the roadmap for CMOS would lead to 3-D devices, such as fin field-effect transistors (FinFETs) and *silicon nanowires* (SiNW). The cross-section of the latter evolved from circular to oval-shaped wires, called *nanosheets* [5], today used by two of the top manufacturing companies at the 3-nano node and beyond. The fabrication of SiNW with undoped channels is attractive, as it simplifies manufacturing and avoids *variability* problems, but it enables both types of carriers to flow (i.e., ambipolar conduction), which may be disruptive if not controlled. By creating devices with an additional *polarity gate* that affects the silicon–metal contact Schottky junctions, it is possible to suppress selectively a type of carrier and to realize electrically controllable N or P devices [7], [8]. This technique is called *electrostatic doping*, and it can be applied after fabrication to reconfigure devices or during operation, to unlock a new functionality (Figure 1). The use of such *controlled-polarity devices* enabled us to create new transistor configurations. It is important to remark on the design symmetry in achieving NANDs and NORs and the compactness of the XOR gate that can be realized out of the same template [9]. Moreover, compact 2- and 3-input XORs as well as *majority* functions can be realized with few transistors and the same template (Figure 2).

For decades, digital designers have been taught to realize circuits in terms of NANDs, NORs, and their extensions, that is, AND–OR–inverter gates. Standard design tools privilege the use of libraries based on these premises. A disruptive innovation step was to
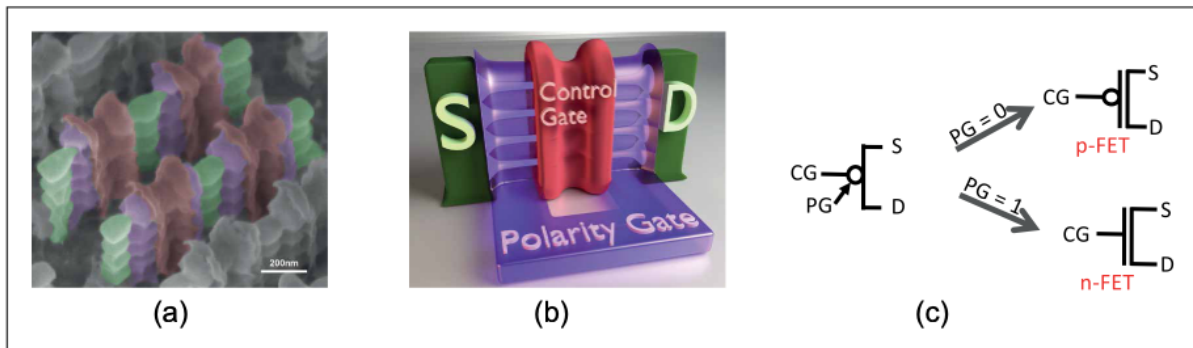
Figure 1. (a) Array of polarity-controlled transistors. (b) Functional sketch of the polarity-controlled transistor. (c) Equivalent schematics.
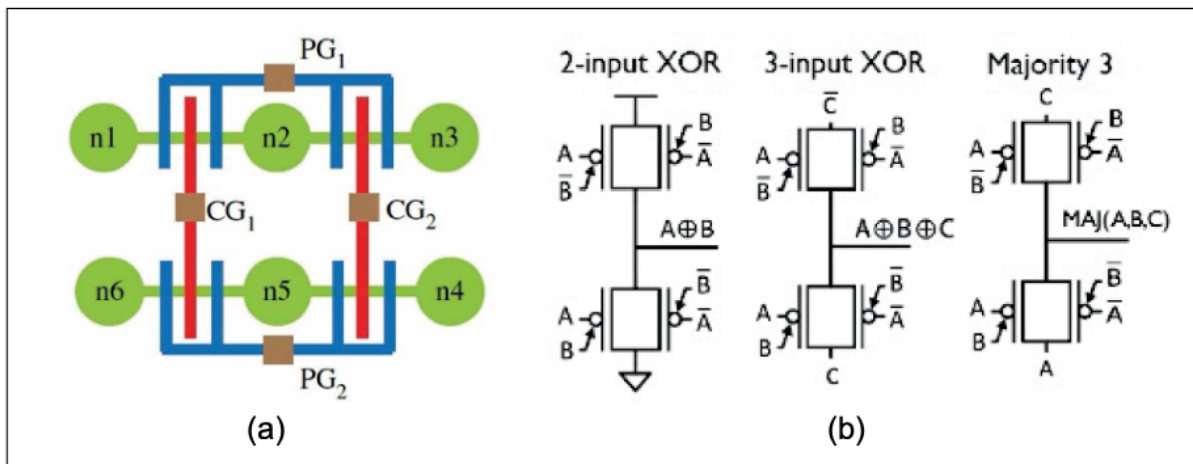

Figure 2. (a) Stick diagram of a tile that can be configured to realize NAND2, NOR2, XOR2-3, or MAJ-3 gates. (b) Transistor interconnection to realize XOR2-3 and MAJ3 gates.

create tools that use the majority gate abstraction, possibly complemented by inverters and/or XOR gates [10]. The algebra based on majority and complementation operations is sound and complete, and therefore algebraic rewriting tools can fully explore the design space, and thus reach the optimum in terms of area and/or delay, even though this process can take over-polynomial time. Logic synthesis tools using the *majority paradigm* have been shown to be applicable in both emerging (e.g., controlled polarity SiNW) [14] and established (e.g., CMOS) technologies. In the latter case, comparisons on standard benchmarks could show a significant improvement in delay (e.g., 15%) in both academic and commercial settings [10].

The lesson to be learned is that emerging technology has spurred interest in new models, algorithms, and tools which eventually proved as an improvement on an established one. This circularity

is captured in Figure 3. By looking at the future, we ask ourselves if the majority paradigm can enable other emerging technologies. The answer is positive, and, for example, some *superconductive technologies* benefit from this approach [11], as the underlying devices have native models based on the majority operation. Similar considerations apply to the emerging domain of optical devices, where interference of waves can be constructive or destructive and the interaction of multiple waves relates to a majority over a set of phase-dependent parameters.

The broad impact of combining research in advancing technology and EDA is well recognized. The term *system-technology co-optimization* (STCO) is often related to designing transistors that will deliver realistic performances in a system. It is a predictive approach, as transistors are used in systems much after their structure and parameters are
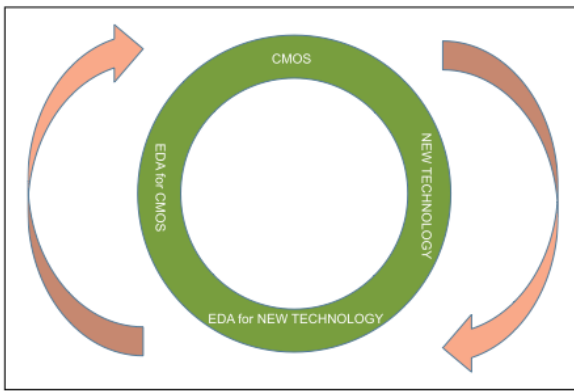
Figure 3. Digital design evolution.

determined. We are now at an inflection point, as it is not clear whether further downscaling beyond the 3-nano node will correlate to silicon area scaling, even though delay and energy reduction are expected. It is my belief that EDA will be the compass to steer among the choices of further downscaling silicon devices, for embracing emerging technologies, or more likely for achieving balanced heterogeneous solutions leveraging specificities of various devices and circuits.

Quantum computing

In the future, the increasing need for computation power will be hardly satisfied by classical computing even with an evolution of technology.

Thus, a disruptive change in the rules of the game is required, as promised by *quantum computing* (QC). A QC can be seen as an accelerator booster connected to a host through an interface, which processes the information in a revolutionary way (Figure 4). QC entails the use of specific emerging technologies (e.g., superconducting electronics, quantum dots, etc.) in combination with a radical change in the computing paradigm. The use of quantum properties of materials to compute, such as *superposition* and *entanglement*, enables computational speed-up at the price of challenging hardware manufacturing and more complex design methods. Current QC implementations require supercooled (e.g., near 0 K) environments or high vacuum, thus making the realization and interfacing of QC hardware expensive and complex. Next, the design technology has to satisfy specific constraints. First, the computation has to adhere to laws of physics, such as *reversibility* and *no cloning*. This implies a different style in designing the combinational logic functional blocks that need to preserve entropy as well as the requirement of not copying quantum states. Second, the design has to leverage the properties and constraints of the QC substrate at best. At the time of this writing, there is a plurality of materials and means to perform quantum computation, and there is not yet a consensus on the "best" approach. Possibly, there may not be even one technology of choice in the
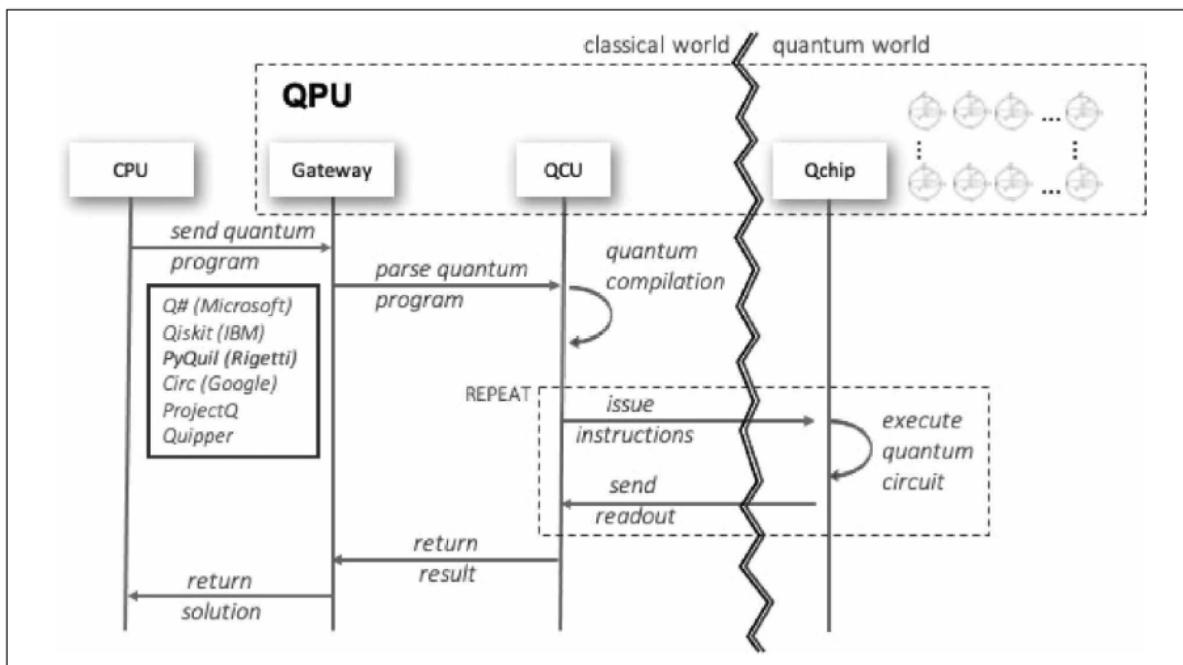

Figure 4. Organization of a quantum computer and interface with the host.

future, as various embodiments will address specific needs, and the competitive advantage will stem from the diversity of solutions. Nevertheless, QC will not scale up and flourish without a design methodology and related EDA tools, which need to be redesigned almost from scratch for QC.

EDA for QC can be split into technology-independent and technology-dependent methods and tools. The former involves *quantum compilation,* where the technology of choice is expressed by a library of quantum gates and their parameters. The latter involves the physical design of the QC chip and its interface to the host computer.

Quantum compilation flows have some similarities to those of conventional computers and processors. First, designers need to express the computational objective with models that go beyond formulae and diagrams, that is, by quantum computation languages that support some type of verification, via simulation or other means. There are specialized design languages for QC, for example, IBM's *Qiskit,* Microsoft's *Q#,* and Google's *Cirq.* Computational models in these languages need to be compiled next into *quantum circuits,* which are a common technology-independent abstraction of an equivalent "assembly code" for quantum computers. Quantum circuits express the computation by a sequence of steps, each one represented by a quantum gate. Quantum circuits are often represented by 2-D array diagrams, with rows corresponding to *quantum bits* (qubits) including *helpers* (also known as *ancillae* that store intermediate computation), and with columns representing the flow of computation (Figure 5). In general, the number of qubits is limited by the physical design of the hardware substrate, while the number of computational steps is limited by the coherence time. The design of optimal quantum circuits entails the mapping of a model of computation expressed in a QC language into quantum circuits
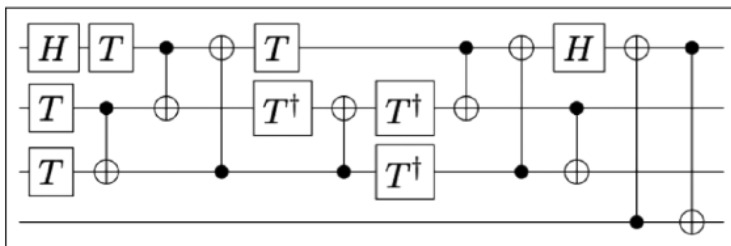
satisfying the technology constraints (i.e., qubits count and coherence time limit) where quantum gates eventually are chosen from a given library, such as Clifford+T. The cost (e.g., execution speed) of each quantum gate depends on the realization technology of choice.

Another class of design steps relates to the physical design of the QC chip and interfaces. In the case of solid-state realizations, physical design involves the classical problems of placement and routing, with additional constraints (e.g., *coupling constraints* to enable 2-quibit operations). Interface design involves mixed-signal and radio-frequency design, to provide means of interacting with the QC substrate. The quantum circuit is interpreted as a sequence of stimuli and the reception of their responses to and from the QC substrate.

At present, good examples of *noise-intermediate quantum* (NISQ) computers have been realized. These systems assume the unavoidable presence of noise in quantum computation, and solutions are applicable to many modeling problems in physics and chemistry not requiring bit-level precision. Today, quantum error detection and correction are subjects of ongoing research, and current solutions are still onerous in terms of qubit usage to achieve precision. Thus, we are not ready yet to put into practice algorithms, such as Shor's algorithms for factoring, that would heavily impact computer science and security in particular.

The role of EDA in QC design is essential for the exploration of future circuits and architectures. Tools and flows are available for quantum compilation, and, in particular, for *reversible logic* circuits and mapping [15], [16], [17] (Figure 6). Nevertheless, the design of QC interfaces and the realization of QC error correction schemes are still not automated. As a result, current QC design platforms still need updates and extensions, thus making EDA for QC a young and vibrant area for research.

It is interesting to note that QC has direct applications to EDA, even though most challenges in EDA relate to combinatorial optimization problems that require quantum computers with error correction hopefully available later in this decade. Nevertheless, exact solutions to most relevant problems would still require over-polynomial time and space, albeit the computational burden, and time may be lower when compared to conventional computing. The application of Grover's algorithm, which speeds

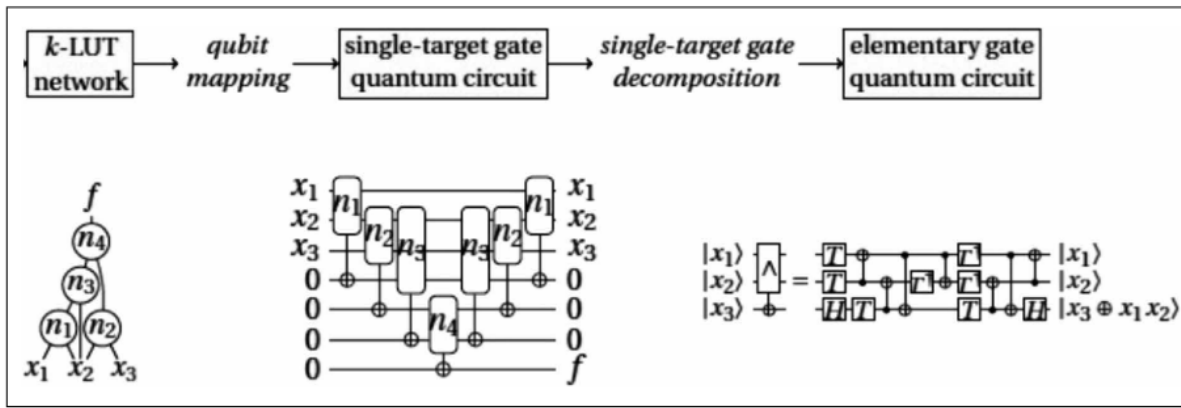**Figure 5. Example of the quantum circuit.**

Figure 6. Steps in reversible logic synthesis: the logic function, the quantum circuit in terms of single-target gates, and the quantum circuit mapped to the Clifford+T library.

up searching quadratically, can find applications in many EDA tools.

Overall, quantum computers can be seen as accelerators that speed up the solution of problems with specific characteristics that can benefit from superposition and entanglement. Present NISQ computers support the analysis of physical phenomena and materials, as they are well-geared to study the evolution in time of Hamiltonian models. Such materials can be used then for future generation QCs. This gives us a path for a cyclic evolution of quantum computers with increasingly better performances. To guide this evolution, EDA is key to determining the useful matches between quantum devices and architectures, that is, determining predictive models of quantum devices to be used in upcoming quantum architectures as well as requirements on material properties, shapes, and composition rules to realize effective quantum hardware devices.

Computing and life

The ultimate question is whether living organisms and computing systems will merge. While an answer is premature and requires broad scientific and philosophical arguments, some converging trends can already be observed. First, biology and medicine have become more quantitative. We can measure many biological processes related to life through the use of sensors. Algorithms for data analysis provide us with profound insights into biological processes. Second, computing systems support *precision medicine* by enabling advanced treatment and new means to deliver health care. Examples include smart drug dispensing and smart robotics,

such as artificial limbs. The partial, but ever-increasing observability and controllability of living organisms is a key aspect of the convergence of life and computing.

One of the most challenging tasks is to understand the brain, this wonderful natural computer that performs many complex functions with only 30 W of power. The brain has inspired new computational paradigms, some of which are extremely powerful as accelerators. *Neuromorphic* computing, using established and/or emerging technologies [13], has shown to be very powerful in solving efficiently some tasks as well as acting as an accelerator. Researchers have also grown neural cells on silicon and created local communication means, as an experiment for joint information processing.

As we humans have always strived for improving ourselves and our conditions, the use of computational devices as assistants has become prevalent to improve our working and social skills. Smart terminals (e.g., phones and watches) can be viewed as common enhancers of our mental skills, for example, assistants to memorize information. The use of these devices has led to changes in our mental functions, with a reduced ability in memorizing data and in handwriting, but with a higher ability in learning and connecting to others. Current personal devices are still loosely connected to us. We acquire information through vision, and we communicate bilaterally with machines through sound or tactile means. New visors for augmented and virtual reality (AR/VR) will deepen the ability and speed to interface with computers and networks through visual and acoustical stimulation. EDA has shown to be essential

not only to designing cutting-edge systems, such as those used in AR/VR and neuromorphic computing. Moreover, EDA tools have been developed and used for codesign of electronics and transducers, such as smart sensors for medical implants.

Medical implants realize a tighter connection between computational platforms and living matter, by using a wide variety of transducers and actuators. For example, pacemakers communicate with the heart via electrical signals, and artificial limbs are likewise electrically stimulated. Electronically controlled drug dispensers (such as insulin pumps) communicate to the body via chemical injection. Some of these systems work in a closed loop, where the information acquired by sensors (e.g., glucose sensors) is processed digitally to control dispensing (e.g., insulin pump). Enormous progress has been achieved in restoring locomotion after spinal cord injury. Understanding and decoding the brain signals is key in providing the appropriate stimulation. Very recent results (May 2023) have reported restoring autonomous locomotion in a human subject through the use of a brain implant connected to an epidural electrical stimulator [19]. This fantastic result has shown how computing can bridge two biological subsystems that were separated by an injury.

Today, medical implants are reserved to correct physical conditions. It is conceivable that in the future electronic devices, with tight integration with biological functions, will be used to enhance our capabilities. In this case, a merger of computing and life will take place, even though possibly in restricted domains. Another loop will close itself. Humans invent technology that can be absorbed by us to enhance the human species itself. The technical feasibility of human enhancement and the ethical evaluation of this step will be one of the most important moments of human evolution.

SCIENCE AND ENGINEERING evolved through centuries as a continuous sequence of steps that can be captured by the motto *natura non facit saltus* (nature does not leap). The rapid changes in the last hundred years are rooted in the discoveries of quantum physics that enabled us to develop semiconductor technology and modern computing. The essence of quantum physics is described as *natura facit saltus* (nature does leap). The ability to disrupt the regular evolution of computing with new materials and paradigms, such as QC, opens new possibilities for solving problems with unprecedented complexity. Similarly, the possibility of harnessing biological systems with artificial computing units paves the way for a new evolution of mankind. ∎

◼ References

[1] A. Sangiovanni-Vincentelli, "The tides of EDA," *IEEE Design Test Comput.*, vol. 20, no. 6, pp. 59–75, Nov. 2003.

[2] D. Hofstadter, *Godel, Escher, Bach: An Eternal Golden Braid*. New York, NY, USA: Basic Books, 1979.

[3] G. Hills et al., "Modern microprocessor built from complementary carbon nanotube transistors," *Nature*, vol. 572, no. 7771, pp. 595–602, Aug. 2019.

[4] G. M. Marega et al., "Low-power artificial neural network perceptron based on monolayer MoS_2," *ACS Nano*, vol. 16, no. 3, pp. 3684–3694, Mar. 2022.

[5] P. Ye, T. Ernst, and M. V. Khare, "The last silicon transistor: Nanosheet devices could be the final evolutionary step for Moore's law," *IEEE Spectr.*, vol. 56, no. 8, pp. 30–35, Aug. 2019.

[6] J. Cong et al., "Accelerator-rich architectures: Opportunities and progresses," in *Proc. Design Automat. Conf. (DAC)*, 2016, pp. 1–6.

[7] M. De Marchi et al., "Polarity control in double-gate, gate-all-around vertically stacked silicon nanowire FETs," in *IEDM Tech. Dig.*, San Francisco, CA, USA, Dec. 2012, pp. 8.4.1–8.4.4.

[8] A. Heinzig et al., "Dually active silicon nanowire transistors and circuits with equal electron and hole transport," *Nano Lett.*, vol. 13, no. 9, pp. 4179–4181, 2013.

[9] S. Bobba and G. De Micheli, "Layout technique for double-gate silicon nanowire FETs with an efficient sea-of-tiles architecture," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 23, no. 10, pp. 2103–2115, Oct. 2015.

[10] L. Amarù, P. Gaillardon, and G. De Micheli, "Majority-inverter graph: A new paradigm for logic optimization," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 35, no. 5, pp. 806–819, May 2016.

[11] G. D. Micheli, "The emerging majority: Technology and design for superconducting electronics," *IEEE Design Test. Comput.*, vol. 38, no. 6, pp. 79–87, Dec. 2021.

[12] M. M. S. Aly et al., "Energy-efficient abundant-data computing: The N3XT 1,000x," *Computer*, vol. 48, no. 12, pp. 24–33, Dec. 2015.

[13] A. Sebastian et al., "Memory devices and applications for in-memory computing," *Nature Nanotechnol.*, vol. 15, no. 7, pp. 529–544, Jul. 2020.

[14] S. Rai et al., "Utilizing XMG-based synthesis to preserve self-duality for RFET-based circuits," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 45, no. 2, pp. 914–927, Mar. 2023.

[15] G. Meuli, M. Soeken, and G. De Micheli, "Xor-and-inverter graphs for quantum compilation," *NPJ Quantum Inf.*, vol. 8, no. 1, p. 7, Jan. 2022.

[16] R. Wille and R. Drechsler, "BDD-based synthesis of reversible logic for large functions," in *Proc. Design Automat. Conf. (DAC)*, 2009, pp. 270–275.

[17] M. Soeken et al., "RevKit: A toolkit for reversible circuit design," *J. Multiple Valued Log. Soft Comput.*, vol. 18, no. 1, pp. 55–65, 2012.

[18] G. V. Resta et al., "Devices and circuits using novel 2-D materials: A perspective for future VLSI systems," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 27, no. 7, pp. 1486–1503, Jul. 2019.

[19] H. Lorach et al., "Walking naturally after spinal cord injury using a brain–spine interface," *Nature*, vol. 618, pp. 126–133, May 2023.

Note

This article summarizes the keynote invited for the 59th DAC conference available at https://www.youtube.com/watch?v=TPIUBXO6_PA

Giovanni De Micheli is a professor and the director of the Integrated Systems Laboratory at EPFL, 1015 Lausanne, Switzerland. He is a Fellow of ACM, AAAS, and IEEE, a member of the Academia Europaea, and an International Honorary member of the American Academy of Arts and Sciences.

■ Direct questions and comments about this article to Giovanni De Micheli, EPFL, 1015 Lausanne, Switzerland; Giovanni.DeMicheli@EPFL.ch.