System-Level Design for Nano-Electronics

David Atienza[1,2], Shashi Kanth Bobba[1], Massimo Poli[3], Giovanni De Micheli[1], and Luca Benini[4]

[1]LSI-EPFL, 1015-Lausanne (Switzerland),{name.surname}@epfl.ch
[2]DACYA-Universidad Complutense, 28040-Madrid (Spain), datienza@dacya.ucm.es
[3]DII-Universita degli Studi di Siena, 53100-Siena (Italy), poli@dii.unisi.it
[4]DEIS-Universita di Bologna, 40136-Bologna (Italy), lbenini@deis.unibo.it

***Abstract*— Latest fabrication technologies of self-assembly nano-circuits (carbon nanotubes, silicon nanowires, etc.) have deployed bottom-up techniques that reach feature sizes well below 65nm, holding great promise for future large silicon-based integrated circuits. However, new nano-devices intrinsically have much higher failure rates than CMOS-based ones. Thus, new design methodologies must address the combination of device-level error-prone technologies with system integration constraints (low power, performance) to deliver competitive devices at the nanometer scale. In this paper we show that a very promising way to achieve nano-scale devices is combining imperfection-aware design techniques during fabrication with gate defect modeling at circuit level. Our results using this approach to define a *Carbon Nanotube Field-Effect Transistor (CNFET)*-based design flow for nanoscale logic circuits attain more than 3× energy-delay-product advantage compared to 65nm CMOS-based ones.**

I. Introduction

The semiconductor roadmap indicates that technological breakthroughs will still enable CMOS evolutionary improvements in the next five to ten years [1]. However, scaling to enhance performance and density is pushing CMOS technology to its intrinsic reliability limits. Increasing leakage worsens power consumption and prevents voltage scaling, and tunneling effects do not enable multiple trimming options for gate insulators. Moreover, each subsequent submicron lithography node requires more complex and expensive masks, and manufacturing yield spreading is progressively becoming a major issue [2].

Due to the increasing issues of CMOS technology, several emerging technologies [3] (e.g.,*Carbon Nanotube Field-Effect Transistors (CNFETs)*, *Silicon Nano-Wires (SiNWs)*, etc.) have been recently proposed as a promising alternative to assemble future high-performance digital systems with billions of transistors. Nevertheless, defect tolerance has been outlined as a key element to effectively use these technologies [10], [4]. Thus, a completely new set of methodologies to design, manufacture and fabricate nano-electronics are needed. We believe such methodologies will be characterized by combined bottom-up and top-down approaches. On one hand, they need to overcome technological limitations at the device level to create partially reliable system components. On the other hand, they need to exploit circuit-level information to efficiently handle remaining manufacturing problems of fundamental nano-devices to create system components, such as, logic circuits or memories.

In this paper we illustrate the efficiency of mixed design methodologies for nano-electronics through a novel design flow for CNFET-based logic circuits. This flow combines two novel imperfection-aware design schemes during fabrication of gates with CNFET modeling and processing at circuit level, which results in overall reductions of logic gates implementation area and guarantees their error-free behavior. Our experimental results illustrates that this approach enables compact and fully-reliable digital designs with more than 3× energy-delay-product advantage compared to 65nm silicon-based CMOS.

The rest of the paper is organized as follows. In Section II we review the work related to CNFETs manufacturing and nano-electronic design methods. Then, in Section III, we describe the proposed design system-level/circuit-level flow to construct reliable CNFET-based gates implementation. Next, we show our experimental results in Section IV highlighting the area savings and other features with respect to CMOS technologies. Finally, we summarize our conclusions in Section V.

II. Related work

Since top-down downscaling became more challenging for digital circuits in latest CMOS technologies, many efforts were put in bottom-up approaches [3], [8], [5]. The ability to design, fabricate and manipulate molecular-scale devices below the lithographic limits was recently demonstrated [13], [10], [14]. They include one- or two-dimensional devices using molecular switches with large conductivity changes depending on molecular states, as well as very thin nano-devices, enabling excellent integration densities.

Logic gates based on CNFETs, SiNWs and switching cross-points were recently demonstrated [6], [12]. Also, architectural paradigms to build logic circuits using NOR/OR planes [2] or cascaded sequences of OR/NOT planes [11] have been already proposed. Additionally, another type of paradigm based on two-terminal components, called NanoFabrics, was introduced in [8]. Moreover, [9] proposes a possible logic-to-layout tool for *Carbon Nano-Tubes (CNTs)*, assuming the use of standard tools and Spice simulation. However, these approaches do not tackle all imperfection factors of emerging technologies and in particular the new design constraints, variable working probabilities of each type of gate and new design constraints of CNFET-based logic circuits.

1-4244-1378-8/07/$25.00 ©2007 IEEE.

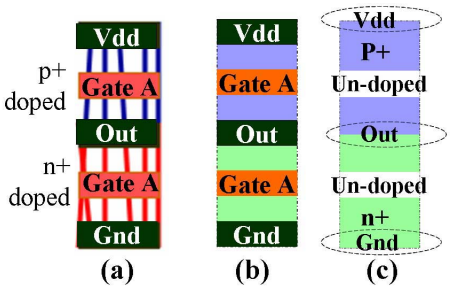

Fig. 1. (a) CNFET-based inverter. (b) Simplified representation of the inverter. (c) Doping profile of the CNTs in the inverter

Besides interconnection and logic, CNFETs and SiNWs have been proposed as fundamental device components of crossbar arrays for information storage due to their regular structure and high density [2]. Also, the information can be stored at the crosspoints in an ideally bi-stable molecular switch, but the integration of molecular switches may suffer from artifacts due to binding between switches and wires [4]. The use of defect-aware system-level design methods to improve synthesis of nanostructures into smaller and reliable blocks, and interfacing modules has already shown the potential of multi-level design methods [7]. Also, design approaches based on reconfiguration around defects to produce nanoscale defect-tolerant digital systems have been proposed [11]. However, these techniques without support at system level produce large area overheads that limit the potential gains of new technologies.

Finally, a major challenge of nanometric technologies is the interface between nano- and micro-scale parts [4], [5], [13]. SiNWs and CNTs can be grown from gold nanocrystals and other materials, and dispersed on the silicon substrate to guarantee addressing properties [5]. However, the need to interface nano- and micro-scale components limits their density and benefits in terms of total available circuit area [11]. Also, in-situ doping during the growth process results in a radial or axial doping profile, which can be exploited to differentiate nanostructures with suitable decoding schemes [13]. Nevertheless, all these approaches are applicable only in particular manufacturing conditions. Thus, the development of reliable and general interfacing schemes between nano- and micro-scale components is still an open research area.

III. Design Framework for CNFET-Based Logic Circuits

In this section we describe the different components that conform our overall design flow for CNFET-based logic circuits. First, we present an example of a simple CNFET-based inverter. Then, we describe two schemes to build CNFET-based logic gates. Finally, we summarize our design flow that exploits the synergy between system- and device-level design to construct reliable and compact nano-scale systems.

A. CNFET-Based Inverter

Figure 1.a shows the actual layout realization of an inverter using CNFET technology, whereas Figure 1.b depicts a simplified representation of the inverter that we use in the rest of the paper. In the used CNFET technology [6], CNTs under the gate region form a channel between the source and the drain when the transistor is in the ON state. Then, Figure 1.c shows the doping profile of the CNTs in the layout cell. The contacts (*Vdd*, *Out* and *Gnd*) are connected to the segments of CNTs where they are doped with n+ or p+ impurities. Similar to a CMOS inverter, the p-CNFET forms the *Pull-Up Network (PUN)* and the n-CNFET forms the *Pull-Down Network (PDN)*. On one hand, a low voltage applied to Gate A makes the undoped CNTs, under the p-CNFET in the PUN, conduct. On the other hand, when high voltage is applied to Gate A, the n-CNFET in the PDN conducts.

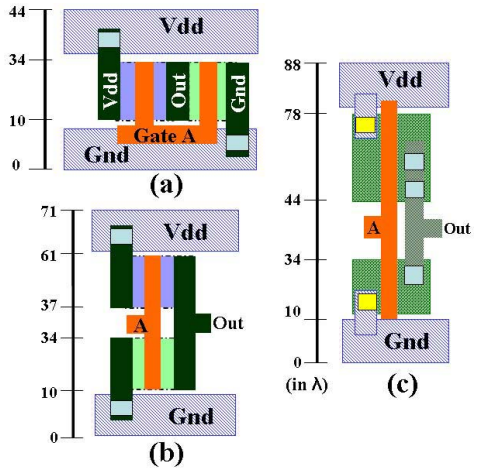

Fig. 2. Inverter realized using Scheme 1 (a), Scheme 2 (b) and 65nm CMOS technology (c)

B. CNFET Layout Schemes

The two major technology challenges for a stable realization of CNFET technology are (i) to avoid metallic CNTs, and (ii) to have a complete control over the growth of CNTs on a substrate; thereby, it avoids misalignment of the CNTs [10]. Regarding the latter challenge, [6] is the first effort to our knowledge that describes a method to handle the problem of CNTs misalignment for logic circuit design in a holistic way, by introducing CNFET layout models that are functionally immune. However, this approach implies area overheads due to insertion of undoped regions in the layout to prevent misalignments. In this paper we present two novel layout schemes that built upon the fundamental concept of defining immune layouts by construction, while avoiding the need of any extra undoped region in the layouts. Figure 2 illustrates the use of our two CNT-misaligned-immune layout schemes to implement an inverter, as realization of the abstraction of the layout scheme shown in Figure 1. Intra-cell routing can be performed in both cases with two metal layers. Figure 1 shows the normalized layouts sizes using a common λ-based scale for clarity purposes.

Figure 2.a shows an inverter realized using Scheme 1, which is a completely new layout proposal with respect to CMOS-based layouts (see Figure 2.c). In this scheme the PUN and

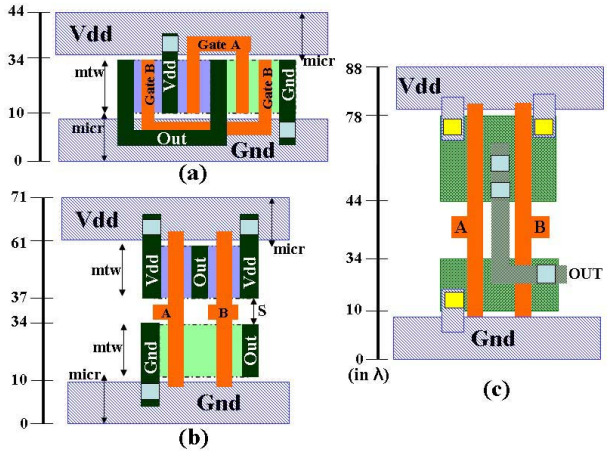

Fig. 3. Layouts of a 2-input NAND realized using Scheme 1 (a), Scheme 2 (b) and 65nm CMOS technology (c)

the PDN are placed next to each other; thereby, the scheme reduces the distance between Vdd and Gnd, which determines the height of standard cells. Then, Figure 2.b shows an inverter realized using Scheme 2. In this case, the PUN is located on top of the PDN, separated by a certain distance, in a similar way as current CMOS technology. Nonetheless, this separation is determined by technology (i.e., 10 λ for 65nm) whereas in Scheme 2 the minimum distance between the PUN and PDN is just limited by litography (i.e., 3 λ for 65nm).

Then, Figure 3 depicts how each scheme can be used to realize a 2-input NAND gate. Interconnection of the two output terminals (*Out*) is not shown in Figure 3.b, but it can be realized in one of the two metal layers. The layouts sizes are normalized using a common λ-based scale for clarity purposes.

Each scheme presents different advantages. Hence, the designer has to select the most appropriate one according to the constraints of each particular gate design. On one hand, Scheme 2 provides minimal use of polysilicon for gating, thereby reducing Ohmic losses. Also, it occupies lesser area as intra-cell routing is much simpler than in Scheme 1. On the other hand, Scheme 1 achieves gains in the height of the standard cells that can be used with this nano-scale technology, which is one key feature to save overall area in large chips. Moreover, Scheme 2 loses its area savings if the Euler path for the PUN and PDN cannot be found having the same sequence of gates, since it would imply to leave CNT gaps using edging minimum spacing.

When compared to CMOS-based layouts, both schemes are more compact as no restriction exists in the placement of the PUN and PDN. Furthermore, the minimum distance between the two networks is just limited by lithography (distance S in Figure 3), while CMOS requires maintaining a certain additional distance between the n- and p-diffusion. To analyze the cell height using 65nm CMOS technologies, the two main parameters to consider are *Maximum Intra-Cell Routing (MICR)* and *Maximum Transistor Width (MTW)*, as indicated in Figure 3). Thus, using 65nm CMOS, MICR is 10λ, S is 3λ and MTW is 24λ; thus, the standard cell size is reduced

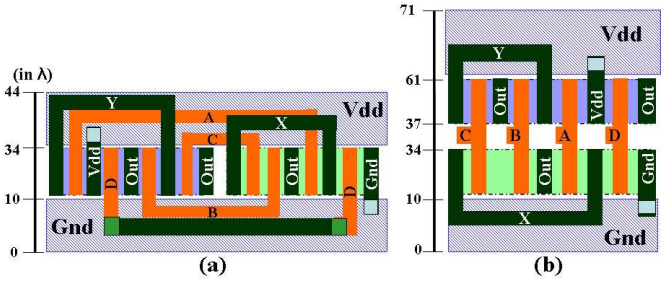

Fig. 4. Examples of layouts for a 2-level complex logic cell:*NOT((A+BC)D)*, realized (a) using Scheme 1 and (b) using Scheme 2. *X* and *Y* are the intermediate metal contacts

by approximately 50% and 19% employing Scheme 1 and Scheme 2, respectively, in comparison to commercial 65nm CMOS cells.

However, in the case of more complex gates, Scheme 1 would require 16λ for MICR due to the complex intra-cell routing. Hence, Scheme 1 achieves 36% gain in standard cell height. An illustration of both schemes to build complex library cells, Figure 4 depicts a representative example to implement of two-level complex logic gate, as combinations of AND, OR and inverters, which are typically included in standard cell libraries.

C. Overall Design Flow

Our design flow to create CNFET-based logic circuits from input Boolean functions is shown in Figure 5. In the first phase, the Logic Optimizer tool takes the original input Boolean function to implement and obtains the corresponding optimized one. This logic optimizer comprises of a logic minimizer tool (e.g., MIS or SIS tool [15]) and generates the final optimized function according to the *Design Constraints (DC)* and the *User Specifications (US)*, such as, redundancy degree of logic as a solution for Metallic CNTs, maximum number of variables in each minterm, etc. To this end, the input DC information is a subset of the *Design Libraries (DL)* specification, which defines the limitations of the underlying CNFET technology to decompose the Boolean logic (e.g., maximum number of variables in a minterm). Regarding the US, it has the data/constraints set by the user (e.g. critical path or power consumption). In the second step, the optimized Boolean function is used as input to the Layout Synthesis Tool; thereby, this tool can generate the layout file. The layout synthesis tool chooses appropriate layout models from the Design Library based upon the US set.

In the following step, the layout file is verified in the Verification Tool. Thus, this step shows the main difference with current state-of-the-art CMOS-based flows, which refers to how new CNFET-based layouts in nano-electronic logic circuits can be funcionally validated. Furthermore, the reliability analysis of the underlying nano-scale technology and its fundamental logic blocks needs to be propagated to enhance the Logic Optimizer; thus, it can initially prune unreliable logic gates decompositions of the input logic functions. Figure 5 shows the three stages of the verification step and the

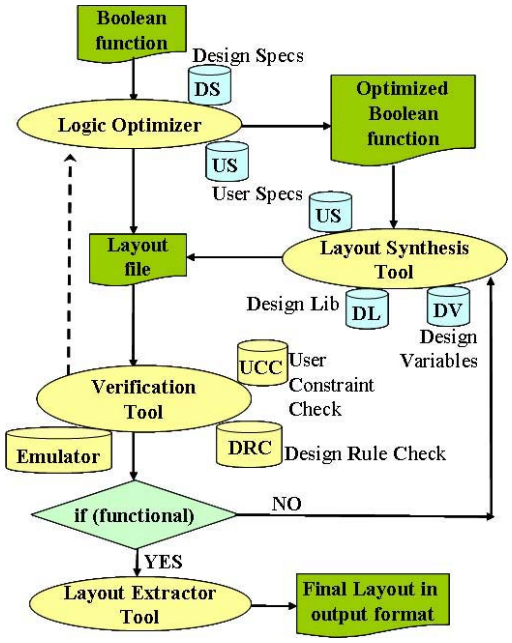

Fig. 5. Overview design flow

propagation of its reliability analysis results to the Logic Optimizer (dotted-line arrow). First, the *User Constraint Check (UCC)* tool verifies if the layout meets the user constraints. Second, the *Design Rule Check (DRC)* engine verifies that the layout is feasible for the underlying CNFET-based library. Currently, all design rules of state-of-the-art 65nm and 32nm process technology are already included in the DRC engine to enable the realization of CNFET-based gates with current technology nodes. The DRC engine is the first element that needs to be changed for different process technologies of nano-electronics. In particular, in our proposed design flow it needs to include the geometric and electrical properties of CNFETs, and how the integration with state-of-the-art CMOS tools can be performed.

Next, our flow proceeds with the extraction of the Boolean function of the layout of the designed circuit and comparing it to the target optimized one. This validation method is the second part that implies a main difference in emerging nano-electronics with respect to current CMOS, due to the high variability and unrealiable manufacturing process of these new technologies, which needs to be modelled by extensive simulation for each target nano-scale manufacturing technology (e.g., CNFETs [11], [6], SiNWs [12], etc). Thus, in our flow we have developed our own CNT Emulator (*Emulator* in Figure 5), which emulates the CNT growth pattern over a substrate. Since CNTs grow in a random fashion [10], the user can model the main parameters that influence the CNTs growth process [5], namely, starting angle, straight length, curving angle, diameter and spacing between CNTs. Then, the user can model a large variety of CNTs growing modes by selecting Flat, Gaussian or Poisson distributions for each CNT parameter. Our emulator applies Monte-carlo simulations to check the layout functionality for the different design points. In case of functional defects, the *Design Variables (DV)* inside the DL are tweaked and the last two steps are iterated until a functional layout is achieved. In the final step, our Layout Extractor Tool produces the final CNFET-based layouts in a format compatible with standard commercial tools (e.g., GDSII [16]), which achieves further integration of our designs in traditional logic circuit manufacturing flows.

IV. Experimental Results

In our first set of experiments we have compared the area employed by our schemes with respect to 65nm CMOS technology layouts. We have taken as case study a 4-input NAND, which is a large logic gate in terms of signal inputs. The 4-input NAND is decomposed into 2 NAND gates, 1 NOR gate and an inverter. In this case, the two proposed layout schemes show an almost equivalent area since the larger building block is a 2-input NAND. Hence, we show in Figure 6 the area of the 4-input NAND in 65nm CMOS technology and using Scheme 2. As this figure shows, the increase of the fan-out of the 4-input NAND increases the area in CMOS and CNFET technology, because larger transistors are required to drive more gates. However, our CNFET-based layouts use smaller area; thus, achieving more than $3\times$ energy-delay-product improvements, while assuming the basic increase of transistors widths of 65nm CMOS technology. Two main reasons exist for the area savings:

1) CMOS layouts need a minimum distance between the n- and p-diffusion area, whereas our layouts developed using Scheme 2 only need a distance of 3λ between the PUN and PDN.
2) The width of CMOS-based p-mos transistors is more than n-mos transistors, whereas CNFET-based transistors have the same widths.

Moreover, although our CNFET-based layouts have assumed equal transistor sizes using a conservative approach for comparison purposes, due to conductivity properties of CNTs [5], they can drive more current and, thus, more gates than respective CMOS models, as Figure 6 shows.

Our second set of results analyses the reliability behavior of various CNT misaligned non-immune layouts of different complexities in 65nm technology. We have included in our study a 3-input NAND, a 2-input NOR and two variations of two-level complex logic gates of similar complexity to the example depicted in Figure 4, which are typically included in standard cell libraries. Then, the results for each type of gates using our CNT-validation step are shown in Figure 7. The different results correspond to various configurations of the emulated CNTs, where a Flat distribution is utilized for CNT-diameter and CNT-spacing, a Gaussian distribution is employed for CNT-starting angle, and either Gaussian or Poisson distributions are used for the curving angle and straight length, as outlined in state-of-the-art CNFET technologies [10], [5].

These results show that the working probabilities for each type of cell varies significantly between different configurations of CNT growth parameters, within values close to 100% to only 65%. Moreover, our results with 2-level logic

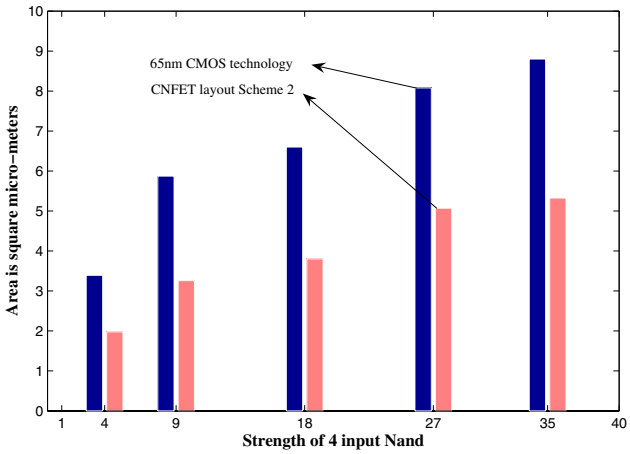

Fig. 6. Area results for a 4-input NAND logic gate using Scheme 2 and CMOS 65nm technology

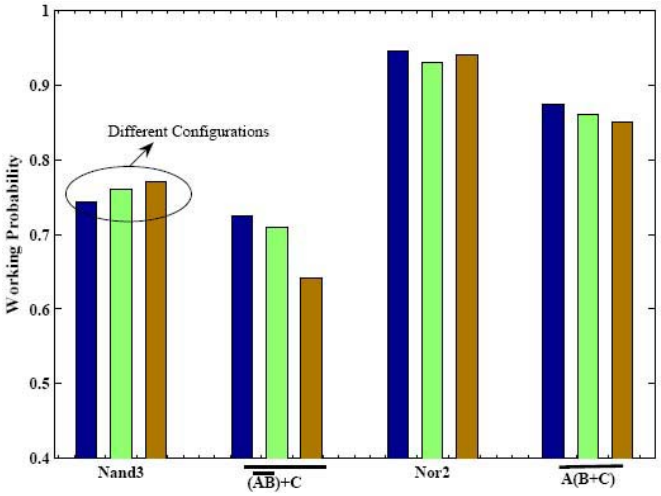

Fig. 7. Analysis of working probabilities for different nano-scale library cells using different 32nm CNFET-based technologies in our logic design flow

cells of similar layout design complexity show important variations (approximately 15%). In fact, even within the same 2-level logic gates layout, working probabilities vary up to 8% for different distributions of CNT growth parameters, as shown in the results of the `NOT(NOT(AB)+C)` logic cell in Figure 7. Furthermore, this variability in working probability increases as more logic levels are added to each basic cells. Thus, the best selection of logical gates to implement a certain Boolean function has a very strong dependency on the concrete features of the used CNFET technology and the logic analyzer must be configured accordingly to choose the right optimized Boolean function. As a result, to guarantee large error-free nano-electronics in an affordable exploration time, new design flows for nano-electronics need to be imperfection-aware, as we propose in this paper; thus, the verification of working probabilities for final layouts can be performed in an automated way inside the tools at the same time as validating traditional design metrics (e.g., design area, speed, etc.) in current CMOS-based circuits.

V. Conclusions

Due to the increasing difficulties to effectively build large integrated circuits in latest CMOS technology nodes, several self-assembly based nano-scale technologies (e.g., CNTs, SiNWs, molecular electronics, etc.) have been proposed to replace CMOS as effective solutions, due to their high integration density and electrical properties. However, these emerging nano-technologies intrinsically have much higher failure rates than current CMOS and novel methods that exploit top-down and bottom-up design flows are needed to deliver competitive devices at the nanometer scale.

In this work we have presented a new design flow for CNFET-based logic circuits that combines system- and device-level techniques to enable compact and reliable digital designs with more than 3× energy-delay-product advantage with respect to equivalent designs using industrial 65nm CMOS technology libraries. In the future we intend to exploit the proposed design flow to identify the potential benefits of improving manufacturing parameters in CNFET-based nano-technologies, such that reliable CAD tools can be developed.

VI. Acknowledgments

This work is partially supported by the Swiss FNS Research Grant 20021-109450/1 and the Spanish Government Research Grant TIN2005-5619.

References

[1] Semiconductor Industry Association, Int. Technology Roadmap for Semiconductors (ITRS), `http://public.itrs.net/`, 2006.
[2] M. Butts, A. DeHon, S. C. Goldstein. Molecular electronics: Devices, systems and tools for gigagate, gigabit chips. *Proc. of ICCD*, 2002.
[3] R. Beckman, E. J.-Halperin, Y. Luo, J. E. Green, J. R. Heath. Bridging dimensions: Demultiplexing ultrahigh density nanowire circuits. *Science*, 310(5747):465–468, 2005.
[4] G. F. Cerofolini. Realistic limits to computation - the technological side. *Applied Physics A*, 86(1):31–42, 2007.
[5] H.-S. P. Wong, J. Appenzeller, V. Derycke, R. Martel, S. Wind, Ph. Avouris, Carbon Nanotube Field Effect Transistors Ű Fabrication, Device Physics, and Circuit Implications. *Proc. of ISSCC*, 2003.
[6] N. Patil, J. Deng, H.-S. Philip Wong, S. Mitra. Automated Design of Misaligned-Carbon-Nanotube-Immune Circuits. *Proc. of DAC*, 2007.
[7] C. He, M. Jacome. Defect-Aware High-Level Synthesis Targeted at Reconfigurable Nanofabrics. *IEEE Trans. on CAD*, 2006.
[8] S. Copen Goldstein and M. Budiu. Nanofabrics: spatial computing using molecular electronics. In *Proc. of ISCA*, 2001.
[9] W. Zhang, and N. K. Jha, ALLCN: An Automatic Logic-to-Layout Tool for Carbon Nanotube Based Nanotechnology. *Proc. of ICCD*, 2005.
[10] P.S. Collins, M.S. Arnold, P. Avouris, Engineering CNT and Nanotube Circuits Using Electrical Breakdown. *Science*, 292:706–709, 2001.
[11] J. Deng, N. Patil, K. Ryu, A. Badmaev, Z. Chongwu, S. Mitra, H-S. P. Wong. CNT Circuits: Circuit-Level Performance Benchmarking and Design Options for Living with Imperfections. *Proc. of ISSCC*, 2007.
[12] Y. Huang, X. Duan, Y. Cui, L. J. Lauhon, K.-Ha Kim, and C. M. Lieber. Logic gates and computation from assembled nanowire building blocks. *Science*, 249:1313–1317, 2001.
[13] M. S. Gudiksen, L. J. Lauhon, J. Wang, D. C. Smith, and C. M. Lieber. Growth of nanowire superlattice structures for nanoscale photonics and electronics. *Nature*, 415:617–620, 2002.
[14] T. Rueckes, K. Kim, E. Joselevich, G. Y. Tseng, C.-L. Cheung, and C. M. Lieber. Carbon nanotube based nonvolatile random access memory for molecular computing. *Science*, 289:94–97, 2000.
[15] G. De Micheli. Synthesis and optimization of digital circuits. McGraw-Hill, 1994.
[16] Cadence Design Systems. The GDS-II Stream Format, 2000.