19

Nanoelectronic Design Based on a CNT Nano-Architecture

Bao Liu
Electrical and Computer Engineering Department
The University of Texas at San Antonio
San Antonio, TX, 78249-0669
Email: bliu@utsa.edu

Abstract — Carbon nanotubes (CNTs) and carbon nanotube field effect transistors (CNFETs) have demonstrated extraordinary properties and are widely expected to be the building blocks of next generation VLSI circuits. This chapter presents (1) the first purely CNT and CNFET based nano-architecture, (2) an adaptive configuration methodology for nanoelectronic design based on the CNT nano-architecture, and (3) robust differential asynchronous circuits as a promising nano-circuit paradigm.

1. Introduction

Silicon based CMOS technology scaling has driven the semiconductor industry towards cost minimization and performance improvement in the past five decades, and is rapidly approaching its end (30). On the other hand, nanotechnology has achieved significant progress in recent years, fabricating a variety of nanometer scale devices, e.g., molecular diodes (44) and carbon nanotube field effect transistors (CNFETs) (46). This provides new opportunities for VLSI circuits to achieve continuing cost minimization and performance improvement in a post-silicon-based-CMOS-technology era.
However, we must overcome a number of significant challenges for practical nanoelectronic systems, including achieving some of the most critical nanoelectronic design metrics as follow.

1. Manufacturability. As minimum layout feature size becomes smaller than lithography light wavelength, traditional lithography based manufacturing process can no longer achieve satisfiable resolution, and leads to significant process variations. Resolution enhancement and other design for manufacturability techniques become less applicable as scaling continues. Alternatively, nanoelectronic systems are expected to be based on bottom-up self-assembly based manufacturing processes, e.g., molecular beam epitaxy (MBE). Such bottom-up self-assembly manufacturing processes provide regular structures, e.g., perfectly aligned carbon nanotubes (23). Consequently, nanoelectronic systems need to rely on reconfigurability to achieve functionality and reliability (51).
2. Reliability. Technology scaling has led to increasingly significant process and system runtime variations, including critical dimension variation, dopant fluctuation, electromagnetic emission, alpha particle radiation and cosmos ray strikes. Such variations cannot be avoided by manufacturing process improvement, and is inherent at nanometer

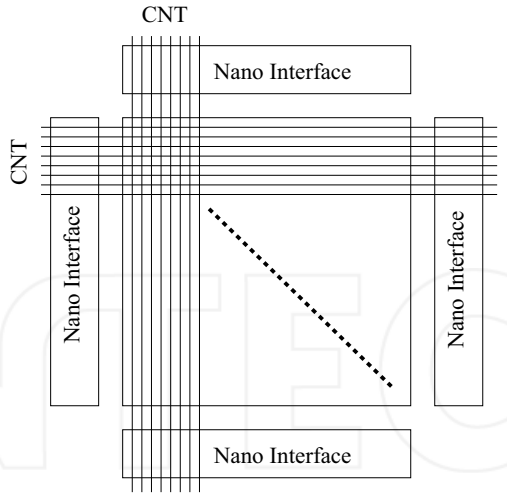

Fig. 1. The proposed CNT crossbar nano-architecture: layers of orthogonal carbon nanotubes form a dense array of RDG-CNFETs and programmable interconnects with voltage-controlled nano-addressing circuits on the boundaries.

scale due to the uncertainty principle of quantum physics. Robust design techniques, including redundant, adaptive, and resilient design techniques at multiple (architecture, circuit, layout) levels, are needed to achieve a reliable nanoelectronic system (5).

3. Performance. Nanoscale devices have achieved ultra-high performance in the absence of load, however, nanoelectronic system performance bottleneck lies in global interconnects. Rent's rule states that the maximum interconnect length scales with the circuit size in a power law (24), while signal propagation delay across unit length interconnect increases as technology scales (30). As a result, interconnect design will be critical to nanoelectronic system performance.
4. Power consumption. As technology scaling leads to increased device density and design performance, power consumption is also expected to be critical in nanoelectronic design.

This chapter presents several recent technical advancements towards manufacturable, reliable, high performance and low power nanoelectronic systems.

1. The first purely CNT and CNFET based nano-architecture, which is constructed by layers of orthogonal CNTs with via-forming and gate-forming molecules sandwiched in between, forming a dense array of reconfigurable double gate carbon nanotube field effect transistors (RDG-CNFETs) and programmable interconnects. Such a CNT array is addressed by novel voltage-controlled nano-addressing circuits on the boundaries, which do not require precise layout design and achieve yield in aggressive scaling and adaptivity to process variations. Simulation based on CNFET and molecular device compact models demonstrates superior logic density, reliability, performance and power consumption for nano-circuits implemented in this CNT crossbar based nano-architecture compared with the existing, e.g., molecular diode and MOSFET based nano-architectures.

2. A complete set of linear complexity methods for adaptive configuration of nanoelectronic systems based on a CNT crossbar based nano-architecture (Fig. 1) (26), including (1) adaptive nano-addressing, (2) RDG-CNFET gate matching, and (3) catastrophic defect mapping methods. Compared with the previous nano-architecture defect mapping and adaptive configuration proposals, these methods are complete, specific, deterministic, of low runtime complexity. These methods demonstrate the promising prospect of achieving nanoelectronic systems of correct functionality, performance, and reliability based on the CNT crossbar nano-architecture.
3. Robust Differential Asynchronous (RDA) circuits as a promising paradigm for reliable (noise immune and delay insensitive) high performance and low power nano-circuits based on the CNT crossbar nano-architecture. Theoretical analysis and SPICE simulation based on 22nm CMOS Predictive Technology Models show that RDA circuits achieve much enhanced reliability in logic correctness in the presence of a single bit soft error or common multiple bit soft errors, and timing correctness in the presence of parametric variations given the physical proximity of the circuit components.

The rest of this chapter is organized as follows. Section 2 reviews the existing nanoelectronic devices, nano-architectures and nano-addressing circuits. Section 3 presents the proposed CNT crossbar based nano-architecture including a novel RDG-CNFET device, a multi-layer CNT crossbar structure, and a voltage-controlled nano-addressing circuit. Section 4 presents adaptive configuration methods for nanoelectronic systems based on the CNT crossbar nano-architecture. Section 5 presents robust differential asynchronous circuits as a promising nano-circuit paradigm. Section 6 presents simulation results which evaluate the CNT crossbar nano-architecture and robust differential asynchronous nano-circuits. Section 7 concludes this paper with a list of nanotechnologies which enable and improve the proposed CNT crossbar based nano-architecture.

2. Background

2.1 Existing Nanoscale Devices

Carbon nanotube is one of the most promising candidates for interconnect technology at nanometer scale, due to its extraordinary properties in electrical current carrying capability, thermal conductivity, and mechanical strength. A carbon nanotube is a one-atom-thick graphene sheet rolled up in a cylinder of a nanometer-order diameter, which is semiconductive or metallic depending on its chirality. The cylinder form eliminates boundaries and boundary-induced scattering, yielding electron mean free path on the order of micrometers compared with few tens of nanometers in copper interconnects (32). This gives extraordinary current carrying capacity, achieving a current density on the order of $10^9 A/cm^2$ (56). However, large resistance exists at CNT-metal contacts, reducing the performance advantage of CNTs over copper interconnects (38).

Among various nanotechnology devices, carbon nanotube field effect transistors are the most promising candidates to replace the current CMOS field effect transistors as the building blocks of nanoelectronic systems. Three kinds of carbon nanotube based field effect transistors (CNFETs) have been manufactured: (1) A Schottky barrier based carbon nanotube field effect transistor (SB-CNFET) consists of a metal-nanotube-metal junction, and works on the principle of direct tunneling through the Schottky barrier formed by direct contact of metal and semiconducting nanotube. The barrier width is modulated by the gate voltage. This device has the most mature manufacturing technique up to today, while two problems limit its

future: (a) The metal-nanotube contact severely limits current. (b) The ambipolar conduction makes this devices cannot be applied to conventional circuit design methods. (2) A MOSFET-like CNFET is made by doping a continuous nanotube on both sides of the gate, thus forming the source/drain regions. This is a unipolar device of high on-current. (3) A band-to-band tunneling carbon nanotube field effect transistor (T-CNFET) is made by doping the source and the drain regions into p^+ and n^+ respectively. This device has low on-current and ultra low off current, making it potential for ultra low power applications. It also has the potential to achieve ultra fast signal switching with $< 60mV/decade$ subthreshold slope (46).
Molecular electronic devices are based on two families of molecules: the catenanes which consist of two or more interlocked rings, and the rotaxanes which consist of one or more rings encircling a dumbbell-shaped component. These molecules can be switched between states of different conductivities in a redox (reduction/oxidation) process by applying currents through them, providing reconfigurability for nanoscale devices (44).
A variety of reconfigurable nanoscale devices have been proposed. Resonant tunneling diodes based on redox active molecules are configurable on/off (44). Nanowire field effect transistors with redox active molecules at gates are of high/low conductance (17). Spin-RAM devices are of high/low conductivity based on the parallel/anti-parallel magnetization configuration of the device which is configured by the polarity of the source voltage (40). A double gate Schottky barrier CNFET is configurable to be a p-type FET, an n-type FET, or off, by the electrical potential of the back gate (25). A double gate field effect transistor with the back gate driven by a three state RTD memory cell is configurable to be a transistor or an interconnect, reducing reconfiguration cost of a gate array (4).

2.2 Existing Nanoelectronic Architectures

At least three categories of nanoelectronic architectures have been proposed. An early nanoelectronic architecture NanoFabrics was based on molecular resonant tunneling diodes (RTDs) and negative differential resistors (NDRs) (20). The insightful authors have observed that passive device (diode/resistor) based circuits lack signal gain to recover from signal attenuation, while combining with CMOS circuits compromises scaling advantages. They proposed latches based on negative differential resistors (NDRs), which, unfortunately, have become obsolete since the publication.
The majority of the existing nanoelectronic architectures are based on a hybrid nano-CMOS technology, with CMOS circuits complementing nano-circuits. In FPNI (50) (CMOL (54)), a nanowire crossbar is placed on top of CMOS logic gates (inverters). The nanowires provide programmable interconnects (and wired-OR logic), while the CMOS gates(inverters) provide logic implementation (signal inversion and gain). Such architectures achieve compromised scaling advantage in term of device density. DeHon (11; 13) proposed to combine programmable nanoscale diode logic arrays with fixed simple CMOS circuitry, e.g., of precharge and evaluation transistors as in domino logic for signal gain. Sequential elements need also to be implemented as CMOS circuits. However, the optimal size of a combinational logic block is typically small (e.g., of 30-50 gates), which results in significant CMOS circuitry overhead in such architectures. An exception is memory design, where CMOS technology provides peripheral circuitry such as address decoders and read sensors with moderate overhead, while nanotechnology provides scaling advantage in memory cells (17; 47; 63).
The third category of existing nanoelectronic architectures rely on DNA-guided self-assembly to form 2-D scuffles for nanotubes (42; 43) or 3-D DNA-rods (14). Such technologies target application in the far future.

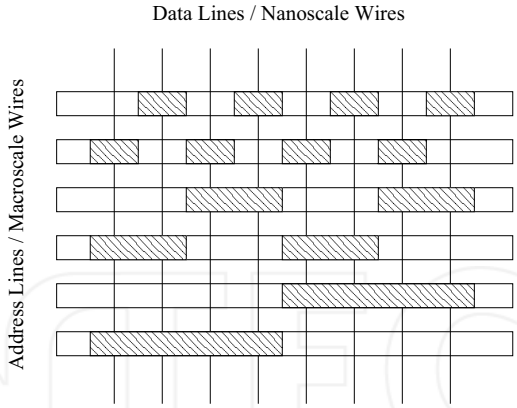

Fig. 2. Layout of undifferentiated nanoscale wires (data lines) addressed by microscale wires (address lines). Lithography defines high- and low-k dielectric regions, which gives field effect transistors and direct conduction, respectively.

2.3 Existing Nano-Addressing Circuits

A nano-addressing circuit selectively addresses a nanoscale wire in an array, and enables data communication between a nano-system and the outside world. The existing nano-addressing circuits are based on binary decoders, with an array of (microscale) address lines running across the (nanoscale) data lines, forming transistors at each crossing (e.g., Fig. 2). Each data line is selected by a unique binary address, given each data line has a unique gate configuration. However, such precise layout design is highly unlikely to achieve at a sublithographic nanometer scale (without significantly compromised yield).
In details, the existing nano-addressing circuits are in four categories as follow.

1. Randomized contact decoder (59) includes gold particles which are deposited at random as contacts between nanoscale and microscale wires. Testing and feedback provide a one-to-one mapping between a nanoscale wire and an address.
2. Undifferentiated nanoscale wires are addressable by microscale wires with (e.g., lithography defined) different gate configurations (which requires nanoscale wire spacing in the same order of lithography resolution) (22) (Fig. 2).
3. Alternatively, different gate configurations are realized in the nanoscale wires, by growing lightly-doped and heavily-doped carbon nanotubes of different length alternatively, while the microscale wires are undifferentiated. A microscale wire crossing a lightly-doped nanotube segment forms a gate, while a heavily-doped nanotube segment is always conductive for all possible signals in the microscale wire. In such a case, precise control of the lengths of the lightly- and heavily-doped nanotube segments would be critical (12; 21).
4. In radial addressing, multi-walled carbon nanotubes are grown with lightly- and heavily-doped shells, an etching process removes the heavily-doped outer shells at precise locations, and defines the gate configurations at each crossing of nanoscale and microscale wires (48).

Because process variations are inevitably significant at nanometer scale, these existing nano-addressing structures achieve limited yield, e.g., there is certain probability that two nanoscale wires have identical or similar gate configuration due to process variation. Furthermore, nanoscale wires are mostly partially selected, e.g., they may not achieve the ideal conductivity upon selected, due to process variations such as misalignment, dopant variation, etc.

2.4 Existing Nano-Architecture Defect-Mapping and Adaptive Configuration Methods

Existing nano-architecture defect mapping techniques are as follow. (1) On a Teramac reconfigurable computing platform, signals are propagated along each row or each column in a crossbar structure, a defect is located at the intersection of a defective row and a defective column, based on the assumption that a single defect is present (10). (2) In the NanoFabrics nano-architecture, the roughly estimated number of defects for a subset of computing resources are collected by counter or none-some-many circuits, a simple graph based algorithm or a Bayes' rule based probabilistic computation procedure gives defect occurrence probability estimates. E.g., highly likely defects are detected in the *probability assignment* phase, which accumulates defect probability in different test configurations, while less likely defects are located in the *defect location* phase, which incrementally clears certain spots as non-defects during test of different configurations (33). (3) A Build-In Self-Test (BIST) method in the NanoFabrics nano-architecture brings much increased complexity with limited applicability (in finding available defect-free neighboring nanoBlocks to implement test circuitry) (8; 58).
After a defect map is achieved presumably, logic circutis can be constructed avoiding or utilizing the defects. For example, a nanoPLA block can be synthesized in the presence of defective crosspoints (36), a CNT nano-circuit layout can be synthesized in the presence of misaligned and mispositioned CNTs (41), metallic CNTs (61), and CNTs of variational density (62).

2.5 Existing CNT Nano-Circuit Design

A very limited number of primitive combinational logic circuits have been fabricated based on CNFETs, including an inverter and two NOR gates in NMOS logic based on SB-CNFETs (3), and a five-inverter ring oscillator based on MOSFET-like CNFETs (9). While nano-circuits based on ambipolar SB-CNFETs need different topologies (3; 46; 53), nano-circuits based on unipolar MOSFET-like CNFETs can be identical to CMOS circuits (9).

3. CNT Crossbar based Nano-Architecture

As we have seen, most existing nanoelectronic architectures are based on diode/resistor logic and CMOS/nano-technologies (11; 13; 20; 50; 54; 63), which only achieve limited manufacturability, reliability, and performance. Carbon nanotubes (CNTs) and carbon nanotube field effect transistors (CNFETs) are the most promising candidates as the the building blocks of nanoelectronic systems due to their extraordinary properties. CNTs possess excellent electrical current carrying capability, thermal conductivity, and mechanical strength. CNFETs are potential to achieve high on-current, ultra-low off-current, and ultra-fast switching ($< 60mV/decade$ sub-threshold slope). CNT crossbar structure (Fig. 1) is one of the most promising candidates for nanoelectronic design platform. Recently, UIUC researchers have achieved fabrication of dense perfectly aligned CNT arrays (23). Such a CNT crossbar structure forms the basis of nanoscale memories (17; 47; 63).
However, no nanoelectronic architecture has been proposed which is solely based on CNTs and CNFETs. The reasons include lack of (1) a reconfigurable CNT based device which could provide functionality and reliability, (2) a self-assembly process which forms complex CNT

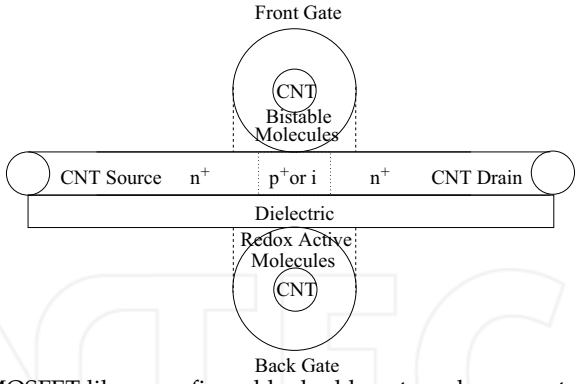

Fig. 3. A n-type MOSFET-like reconfigurable double gate carbon nanotube field effect transistor (RDG-CNFET).

structures, and (3) an achievable mechanism which precisely addresses an individual CNT in an array.
In this section, we investigate the first purely CNT and CNFET based nano-architecture, which is based on a novel RDG-CNFET device, includes a CNT crossbar structure on multiple layers, and a novel voltage-controlled nano-addressing circuit.

3.1 RDG-CNFET Device Structure

As the building block of a purely CNT and CNFET based nano-architecture, a reconfigurable double-gate CNFET (RDG-CNFET) is constructed by sandwiching electrically bistable molecules in a double gate CNFET. The double gate CNFET is constructed by three overlapping orthogonal carbon nanotubes. The top and the bottom carbon nanotubes form the front gate and the back gate, while doping the carbon nanotube in the middle layer forms the source and the drain of a n- or p-type MOSFET-like CNFET (46). Electrically bistable molecules are coated around the front gate and sandwiched between the front gate and the source/drain regions. Dielectric and redox active molecules are coated around the back gate and sandwiched between the back gate and the source/drain regions (Fig. 3).
The redox active molecules at the back gate are electrically reconfigurable to hold/release charge in a redox process, which controls the CNFET threshold voltage and conductance, or, turns the CNFET on or off. An example of such configuration is reported in (17), wherein a $\pm 10V$ voltage applied to cobalt phthalocyanine (CoPc) molecules triggers a redox process, and results in a NW-FET conductance change of nearly 10^4 times. Such reconfiguration of CoPc molecules is repeatable for more than 100 times.
The bistable molecules sandwiched between the front gate and the source/drain regions are electrically reconfigurable to be conductive or insular, making the device a via or a FET. An example of such electrically bistable molecules is reported in (44), wherein oxidative degradation reduces resonant tunneling current of the V-shaped amphiphilic [2]-rotaxane 5^{4+} molecules by nearly a factor of 100. Alternatively, the anti-fuse technologies in the existing reconfigurable architectures provide one-time configurability. For example, the QuickLogic ViaLink technology include a layer of amorphous silicon sandwiched between two layers of metal. A $10V$ programming voltage provides a resistance difference between $G\Omega$ and 80Ω (6).

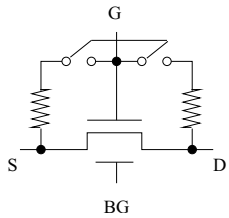

Fig. 4. Compact model of a n-type MOSFET-like reconfigurable double gate carbon nanotube field effect transistor (RDG-CNFET).

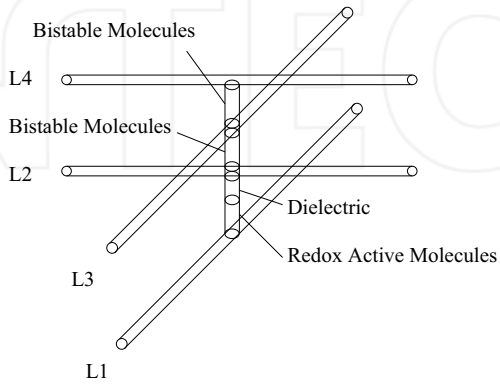

Fig. 5. Carbon nanotube (CNT) layers in the proposed nanoelectronic architecture.

3.2 RDG-CNFET Device Behavior

Such a RDG-CNFET device is described in a compact model as is shown in Fig. 4, and is reconfigurable to the following components, making it an ideal nanoelectronic architecture building block.

1. Via, when the front gate bistable molecules are configured to be conductive. The overlapping of the front gate and the source/drain regions form conductive contacts. As a result, the front gate, the source, and the drain are short circuited. The device is configured as a via between the carbon nanotubes on the top and in the middle.
2. Short, when the front gate bistable molecules are configured to be insular, and the back gate redox active molecules are configured to hold positive(negative) charge in a n-type(p-type) CNFET. The CNFET is on for any front gate voltage.
3. MOSFET-like CNFET, when the front gate bistable molecules are configured to be insular, and the back gate redox active molecules are configured to hold negative(positive) charge in a n-type(p-type) MOSFET-like CNFET. The CNFET threshold voltage is adjustable by the doping concentration in the channel (p or n doping for a n- or p-type CNFET), such that when the back gate redox active molecules are configured to hold negative(positive) charge in a n-type(p-type) MOSFET-like CNFET, the CNFET achieves both performance and leakage control.
4. Open, when the MOSFET-like CNFET is turned off. This is achieved at the architecture level as follows.

3.3 CNT Crossbar Structure

At a larger scale, a nanoelectronic architecture is constructed by growing layers of orthogonal carbon nanotubes, with via-forming (electrically bistable) and gate-forming (dielectric and redox active) molecules sandwiched at each crossing (Fig. 1). The carbon nanotubes are either (1) semiconductive CNTs which are doped to have low resistivity and are reconfigurable to opens by gate isolation, or (2) metallic CNTs which upon identification can be utilized as global interconnects if not avoided or removed (1; 64). The (1) via-forming (electrically bistable) and (2) gate-forming (dielectric and redox active) molecules can be first coated around a carbon nanotube (e.g., as in (17)), then undergo an etching process with the top layer of carbon nanotubes as masks (e.g., as in (49)). The remaining molecules are sandwiched between two orthogonal carbon nanotubes on adjacent layers. A top-down (e.g., lithography) process defines the areas for each type of molecules to assemble on each layer, as well as the p-wells and n-wells. P-type and n-type of MOSFET-like CNFETs are formed by (e.g., potassium or electrostatic (46)) doping of the carbon nanotubes selectively. E.g., a p-well or n-well of dimensions in the order of $22nm$ include about 10 rows of CNFETs.

Configuration of such a CNT crossbar based nanoelectronic architecture gives a nanoscale VLSI implementation including MOSFET-like CNFETs and interconnects with opens, shorts and vias (Fig. 6), which can be 2-D (compatible to traditional VLSI systems) or 3-D.

In a 2-D VLSI implementation, MOSFET-like CNFETs are formed on the bottom three layers of carbon nanotubes, with the first layer (L1) from bottom of carbon nanotubes provides the back gates, the second layer (L2) provides the source and the drain regions, and the third layer (L3) provides the front gates of the MOSFET-like CNFETs. Dielectric and redox active (back gate) molecules are sandwiched between the L1 and L2 layer carbon nanotubes, and electrically bistable (front gate) molecules are sandwiched between the L2 and L3 layer carbon nanotubes. A multi-layer reconfigurable interconnect structure with programmable vias and opens is achieved with via-forming and gate-forming molecules sandwiched between interconnects which are formed above the first (L1) layer (Fig. 5).

3-D VLSI circuits are under active research in recent years due to their potential of achieving reduced wirelength, reduced power consumption and improved performance. However, silicon based VLSI circuits are essentially 2-D, because MOSFETs are surface devices on the bulk of silicon, 3-D MOSFET circuits can only be achieved by bonding chips. It is therefore critical to achieve (1) bonding technology which provides acceptable mechanical strength, (2) via technology which provides low resistive interconnects between chips, and (3) heat dissipation in a multiple chip system for silicon based 3-D circuits. On the contrary, CNFET and CNFET based nano-architectures provide excellent platforms for 3-D VLSI circuits, because (1) CNTs and CNFETs are not confined to certain surface and can be manufactured in 3-D space, (2) CNTs possess excellent current carrying, mechanical and heat dissipation properties which are critical to 3-D VLSI circuits.

In a 3-D VLSI implementation, the RDG-CNFETs do not need to be confined on the bottom layers, with the upper layers dedicated to interconnects. Instead, transistors and interconnects are free to be located on each layer of carbon nanotubes. Gate forming (dielectric and redox active) molecules and via-forming (electrically bistable) molecules are distributed between adjacent CNT layers. Combination of the types of molecules surrounding a CNT segment gives three components.

1. Gate-forming molecules both on top and on bottom of a CNT segment give a device which is reconfigurable to either open or short,

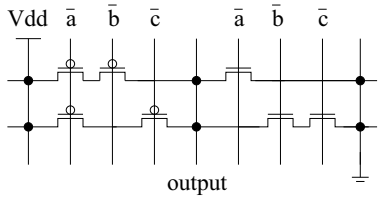

Fig. 6. An RDG-CNFET based Boolean logic $a(b+c)$ implementation.

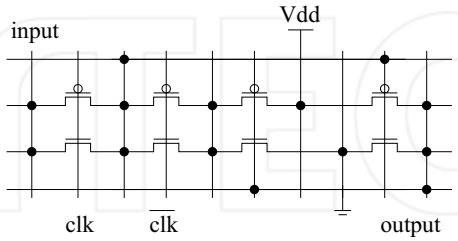

Fig. 7. An RDG-CNFET based latch implementation.

2. Gate-forming and via-forming molecules on top and on bottom of a CNT segment give the RDG-CNFET, which is reconfigurable to via, short, MOSFET-like CNFET, and open,
3. Via-forming molecules both on top and on bottom of a CNT segment give a device which is reconfigurable to be stacked via, simple via, or double gate FET.

We have the following observations.

Observation 1. *Via-forming (electrically bistable) molecules must be present between any two adjacent layers.*

Observation 2. *Gate-forming (redox active) molecules must be present next to each layer for gate isolation.*

Observation 3. *Gate-forming (redox active) and via-forming (electrically bistable) molecules need to be evenly distributed on each layer for performance.*

3.4 Circuit Paradigms and Analysis

This CNT crossbar based nano-architecture provides regularity and manufacturability for high logic density implementations of all CMOS logics, including the standard CMOS logic (e.g., in Fig. 6), domino logic, pass-transistor logic, etc., for combinational circuits, as well as latches (e.g., in Fig. 7), flip-flops, memory input address decoder and output sensing circuits. Such high logic density is achieved via direct connection of CNFETs through their source/drain regions (e.g., as in an latest Intel microprocessor implementation (15)), without going through additional (e.g., metal) interconnects. CNT-metal contacts are known to bring the most significant resistivity in CNT technology (38). Avoiding such CNT-metal contacts contributes to performance and reliability improvements. Furthermore, reduced interconnect length also leads to reduced interconnect capacitance, and improved circuit performance.
This CNT crossbar based nano-architecture also provides a high reconfigurability by allowing an arbitrary ratio of logic gates and interconnect switches (a RDG-CNFET device can be

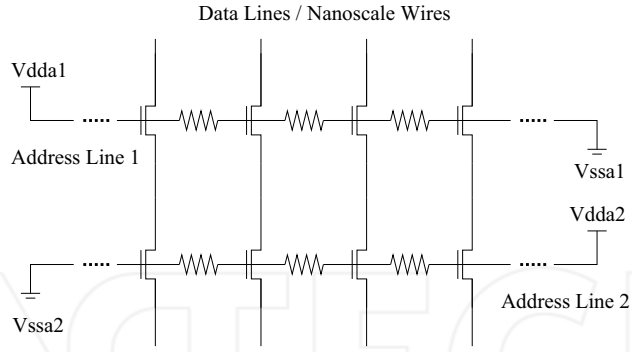

Fig. 8. Schematic of the proposed voltage-controlled nano-addressing circuit.

configured as either a logic gate or an interconnect switch). A pre-determined ratio of logic devices and interconnect switches (e.g., in standard cell designs and FPGA architectures where cells and routing channels are separated) constrains design optimization and may lead to inefficient device or interconnect utilization. Allowing an arbitrary ratio of logic gates and interconnect switches (e.g., as in sea-of-gate designs) provides increased degree of freedom for design optimization (4).

The CNT crossbar based nano-architecture is also the first to include multiple routing layers. Multiple routing layers (as in the current technologies) are necessary for VLSI designs, as Rent's rule suggests that the I/O number of a circuit module follows a power law with the gate number in the module (24). A small routing layer number could lead to infeasible physical design or significant interconnect detouring, resulting in degraded performance and device utilization.

3.5 Voltage Controlled Nano-Addressing Structure

The final piece of the CNT crossbar nanoelectronic architecture is the nano-addressing circuits on the boundary of the carbon nanotube crossbar structure.

Designing a nano-addressing circuit is a challenging task, because (1) the nanoscale layout cannot be manufactured precisely unless it is of a regular structure, and (2) the nano-addressing circuit cannot be based on reconfigurability since it provides reconfigurability to the rest of the nanoelectronic system.

A novel voltage-controlled nano-addressing circuit (Fig. 8) is constructed by running two address lines (of either microscale or nanoscale wires) on top of the data lines (of nanoscale wires in an array which are to be addressed). The address lines and the data lines are orthogonal. At each crossing of an address line and a data line, a field effect transistor is formed by doping the data line into the source and the drain regions while the address line provides the gate of the transistor, with a thin layer of dielectric sandwiched between the gate and the transistor channel. Such field effect transistors have been successfully fabricated based on either nanowires or carbon nanotubes (17; 37; 46).

3.6 Voltage-Controlled Nano-Addressing Principle

The address line provides the gate voltage for the transistors. Each address line is connected to two external voltages at the ends (V_{dda1} and V_{ssa1} for address line 1, V_{dda2} and V_{ssa2} for address line 2). The position of a nanoscale wire in the array gives the gate voltage for the transistor

on the nanoscale wire alone the address line. For example, a i-th nanoscale wire (starting from V_{ss}) in an array of n equally spaced nanoscale wires has a transistor gate voltage

$$V_g(i,n) = \frac{i}{n}V_{dd} + \frac{n-i}{n}V_{ss} \tag{1}$$

in an address line connecting to two external voltage sources V_{dd} and V_{ss}. Here we assume uniform address lines of negligible external resistance (from the first or the last nanoscale wire to the nearest external voltage source).

A transistor is on if its gate voltage exceeds the threshold voltage $V_g > V_{th}$. A nanoscale wire is conductive if both transistors on it are on. Because the two address lines provide an increasing series and a decreasing series of gate voltages respectively, only nanoscale wires at specific positions in the array are conductive. For example, for $V_{dda1} = V_{dda2}$ and $V_{ssa1} = V_{ssa2}$, the nanoscale wire in the middle of the array gets conductive.

In general, to select the i-th data line from the left in an array of n nanoscale wires, the external voltages need to be such that all the transistors on the right hand side of the i-th data line in the first address line are off, and all the transistors on the left hand side of the i-th data line in the second address line are off:

$$\begin{aligned} V_{ga1}(i+1,n) &= (1-\frac{i+1}{n})V_{dda1} + \frac{i+1}{n}V_{ssa1} < V_{th} \\ V_{ga2}(i-1,n) &= \frac{i-1}{n}V_{dda2} + (1-\frac{i-1}{n})V_{ssa2} < V_{th} \end{aligned} \tag{2}$$

3.7 Voltage Controlled Nano-Addressing Analysis

Compared with the existing nano-addressing circuits, the proposed voltage-controlled nano-addressing circuit leads to significant manufacturing yield improvement due to the following reasons.

The existing nano-addressing circuits are based on binary decoders and require every nanoscale wire have a *unique physical structure* to differentiate itself, which is highly unlikely in a nanotechnology manufacturing process - lithography cannot achieve nanoscale resolution, while bottom-up self-assembly based nanotechnology manufacturing processes provide only regular structures. Even at microscale, such a structure is subject to prevalent catastrophic defects and significant parametric variations, which result in low yield.

On the contrary, the proposed circuit consists of only uniform components in a regular structure. Every nanoscale wire has a *uniform physical structure* and is *differentiated by their electrical parameters, e.g., the node voltages*. This scheme avoids any precise layout design and significantly improves yield and enables aggressive scaling of the addressing circuit with the rest of the nanoelectronic system.

Furthermore, let us compare voltage-controlled nano-addressing with the existing binary decoder based nano-addressing mechanisms in terms of addressing accuracy and resolution. These two key quantitative metrics for nano-addressing circuits are defined as follow since such definition is not available in previous publications to the best of the author's knowledge.

Definition 1. Addressing inaccuracy *of a nano-addressing circuit is the offset between the target data line i and the data line j of maximum current.*

$$AI = |i-j| \tag{3}$$

In voltage-controlled nano-addressing, addressing inaccuracy is given by inaccurate addressing voltages from the voltage dividers. Such addressing inaccuracy can be further minimized by adjusting the external voltages to adapt to manufacturing process and system runtime parametric variations. As a result, a mis-addressing is localized, i.e., the data line j of maximum current is not far from the target data line i.
In traditional binary decoder based nano-addressing, an n-bit binary address has n neighboring binary addresses of Hamming distance 1. A 1-bit error could lead to n different mis-addressings. This leads to non-localized mis-addressing and a more significant addressing inaccuracy.

Definition 2. Addressing resolution *of a nano-addressing circuit is the minimum ratio between the on current $I_{on}(i)$ of a target data line i and the off current $I_{off}(j)$ of a non-target data line j (under all conditions, e.g., different inputs and parametric variations).*

$$AR = Min\{\frac{I_{on}(i)}{I_{off}(j)}\} \tag{4}$$

In traditional binary decoder based nano-addressing, the achievable addressing resolution depends on the conductance difference between the target data line and other non-target data lines. There are n non-target data lines with Hamming distance 1 for a n-bit target address, which have similar if not identical conductances. The presence of parametric variations further reduces addressing resolution.
In voltage-controlled nano-addressing, addressing resolution is largely given by the addressing voltage difference between two adjacent data lines. Applying high voltages leads to a number of reliability issues, such as electromigration and gate dioxide breakdown. Carbon nanotubes are highly resistive to electromigration, while new material is needed to enhance reliability for gate dioxide breakdown.
Alternatively, for given gate voltage difference, transistor current difference can be improved by improving the inverse subthreshold slope. However, MOSFETs and MOSFET-like CNFETs are limited to an inverse subthreshold slope S (which is the minimum gate voltage variation needed to bring a $10\times$ source-drain current increase) of $2.3\frac{kT}{q} \approx 60mV/decade$ at $300K$ (46). This requires development of novel devices for larger inverse subthreshold slopes.

4. Adaptive Configuration of Nanoelectronic Systems Based on the CNT Crossbar Nano-Architecture

In this section, we examine a list of nanoelectronic design adaptive configuration methods which cancel the effects of catastrophic defects and parametric variations in the proposed CNT crossbar nano-architecture.

4.1 Adaptive Nano-Addressing

A variety of parametric variations are expected to be prevalent and significant in nanoelectronic systems. Their effects on the voltage-controlled nano-addressing circuit (Fig. 8) are as follow.

1. Global address line resistance variations, e.g., due to uniform width, height, and/or resistivity variations of the address lines, have no effect on the voltage divider hence the addressing scheme.
2. Address line misalignment (shifting) has no effect on the conductances of the data lines.

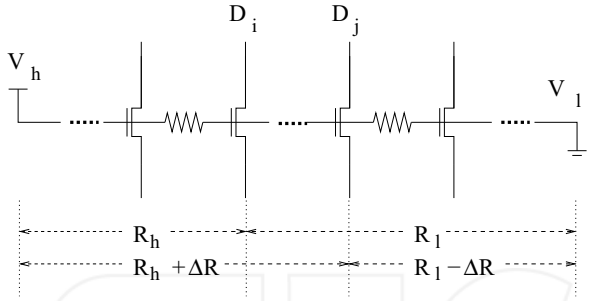

Fig. 9. Addressing two CNTs D_i and D_j with a resistance of ΔR in between.

3. Global data line misalignment (i.e., shifting of all data lines), variations of external voltage sources, and variations of external wire/contact resistance (between the resistive voltage divider and the external voltage sources) lead to potential addressing inaccuracy (CNT target offset).
4. Individual data line misalignment (shifting) could decrease the difference between the gate voltages of two adjacent transistors, leading to degraded addressing resolution (on/off CNT current ratio between two adjacent CNTs).
5. Process variations of the transistors, including width, length, dopant concentration, and oxide thickness variations, lead to transistor conductivity uncertainty and degraded addressing resolution.

The nano-addressing scheme needs to achieve a higher enough addressing resolution which endures the above-mentioned parametric variation effects (e.g., by applying high external addressing voltages, and/or novel CNFETs of $< 60mV/decade$ subthreshold slope).
After achieving satisfiable addressing resolution, we need to minimize any addressing inaccuracy and address the correct CNT data line (Problem 1).

Problem 1 (Adaptive Nano-Addressing)**.** *Given a voltage-controlled nano-addressing circuit, address the i-th CNT data line in the presence of parametric variations.*

Let us first derive the external voltage offset needed for a data address offset. Suppose for an address line, the external voltages V_h and V_l address the i-th CNT data line D_i. The resistance between CNT data line D_i and the high (low) external address voltage V_h (V_l) is R_h (R_l) (Fig. 9).[1] We have

$$\frac{R_l}{R_h + R_l} V_h + \frac{R_h}{R_h + R_l} V_l = V_{on} \tag{5}$$

where V_{on} is the voltage needed to address a CNT data line of peak current. Shifting the external voltages to $V_h + \Delta V$ and $V_l + \Delta V$ addresses another CNT data line D_j. The resistance between CNT data line D_j and the high (low) external address voltage is $R_h + \Delta R$ ($R_l - \Delta R$). We have

$$\frac{R_l - \Delta R}{R_h + R_l}(V_h + \Delta V) + \frac{R_h + \Delta R}{R_h + R_l}(V_l + \Delta V) = V_{on} \tag{6}$$

[1] For the first address line, the high external voltage $V_h = V_{l1}$ is on the left, the low external voltage $V_l = V_{r1}$ is on the right. For the second address line, the high external voltage $V_h = V_{r2}$ is on the right, the low external voltage $V_l = V_{l2}$ is on the left.

As a result,

$$\Delta V = \frac{\Delta R}{R_h + R_l}(V_h - V_l) \tag{7}$$

Observation 4. *The external voltage offset ΔV is proportional to the resistance offset ΔR between two CNT data lines, and is proportional to the physical offset ΔL between the two CNT data lines, if the resistive voltage dividers are uniform (e.g., the CNT data lines are equally spaced and the address lines have uniform resistivity).*

Based on Observation 1, Method 1 gives an adaptive nanoelectronic addressing method, which finds the external voltage shifts needed to address the left most and the right most CNT data lines first. Any other external voltage shift needed to address a specific CNT data line is then computed based on a linear interpolation. To address the left most or the right most CNT data line, we apply a gradually increasing/decreasing external voltage offset ΔV at an address line, keep all the transistors at the other address line on, and measure the conductance of the array of CNT data lines. The maximum and the minimum ΔV's (e.g., ΔV_{mink} and ΔV_{maxk}, $k = 1$ or 2) with non-zero CNT data line conductances address the left most and the right most CNT data lines, respectively.

Algorithm 1: Adaptive Voltage Controlled Nano Addressing

Input: An array of n CNT data lines, address i
Output: Addressing i-th data line

1. Turn on all transistors at address line 2 ($V_{l2} = V_{r2} > V_{th}$)
2. Find ΔV_{l1} which addresses first data line (binary search)
3. Find ΔV_{r1} which addresses n-th data line (binary search)
4. Turn on all transistors at address line 1 ($V_{l1} = V_{r1} > V_{th}$)
5. Find ΔV_{l2} which addresses first data line (binary search)
6. Find ΔV_{r2} which addresses n-th data line (binary search)
7. Shift V_{l1} and V_{r1} by $\frac{n-i}{n}\Delta V_{l1} + \frac{i}{n}\Delta V_{r1}$
8. Shift V_{l2} and V_{r2} by $\frac{n-i}{n}\Delta V_{l2} + \frac{i}{n}\Delta V_{r2}$

Observation 5. *The addressing accuracy given by Method 1 depends only on the uniformity of the resistive voltage divider, and the time domain variations of the external voltage differences $V_{l1} - V_{r1}$ and $V_{l2} - V_{r2}$. Any time-invariant (e.g., manufacturing process) variations of the external voltages (V_{l1}, V_{r1}, V_{l2}, and V_{r2}) or the external address line resistances (from the outer most data lines to the external voltage sources) do not affect the achievable addressing accuracy.*

4.2 RDG-CNFET Gate Matching

Another process variation is the misalignment of the front gate CNT and the back gate CNT of a reconfigurable double-gate CNFET (RDG-CNFET). This is because that the front gate CNT and the back gate CNT are on different ($i - 1$ and $i + 1$) layers, while CNT arrays on different layers do not have and are not expected to have a *precise* alignment mechanism.

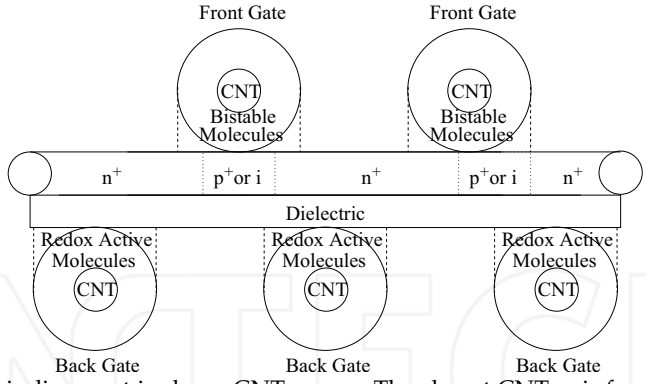

Fig. 10. CNT misalignment in dense CNT arrays. The closest CNT pair forms the front gate and the back gate of a RDG-CNFET. The neighboring CNTs have cross-coupling effect which needs to be simulated/tested or avoided by shielding.

Fortunately, we observe that precise alignment between a front gate CNT and a back gate CNT is not necessarily required as long as the CNT arrays are dense, e.g., with the spacing between CNTs close to the CNT diameters. In such a case, a double gate field effect transistor is formed even in the presence of CNT misalignment (Fig. 10). A CNFET channel is formed by doping the source/drain regions with the front gate CNT on the upper layer as mask. The resultant CNFET channel aligns with the front gate CNT. A misaligned back gate injects a weaker electrical field in the CNFET channel from a longer distance. A neighboring back gate may also injects a weak electrical field in the channel. This is either tolerated (which needs to be verified by simulation or testing) or avoided (by reserving the neighboring back gates for shielding).
The question is then how to find the closest CNT pair on different layers which form the front gate and the back gate of a RDG-CNFET (such that we can address them and configure the RDG-CNFET).

Problem 2 (RDG-CNFET Gate Matching)**.** *Given a CNT i on layer l, locate the closest CNT j on layer $l+2$ (or $l-2$) where CNTs i and j form the front gate and the back gate of a RDG-CNFET.*

Method 2 solves Problem 2 and finds the closest CNT pairs which form the front gate and the back gate of a RDG-CNFET.

Algorithm 2: RDG-CNFET Gate Matching

Input: CNT i on layer l which is a gate of CNFET T
Output: Closest CNT j to CNT i on layer $l+2$ ($l-2$) which is the other gate of CNFET T

1. Apply a turn-off gate voltage to CNT i
2. For each CNT j on layer $l+2$ ($l-2$),
3. Apply a turn-off gate voltage to CNT j
4. Measure the conductance of CNFET T
5. Find CNT j for the smallest CNFET conductance

Once a matching gate is identified, the CNFET can be characterized (by achieving its I-V curves). A parasitic CNFET can also be identified by finding the second closest CNT (with the second smallest CNFET conductance in the algorithm), which is either tolerated or avoided in a nanoelectronic design.

4.3 Catastrophic Defects and Mapping Techniques

In this subsection, we examine catastrophic defects for CNTs, programmable vias and CNFETs, and their corresponding detection and location methods.

4.3.1 Metallic, Open and Crossover CNTs

CNTs are metallic or semiconductive depending on their chirality. One third of CNTs are metallic if they are grown isotropically. Metallic CNTs can be removed by either chemical etching (64) or electrical breakdown (1). However, such techniques bring large process variation effects (34). Mitra et al. propose use of CNT bundles for each nanoelectronic signal to reduce metallic CNT effect (34). We observe that metallic CNTs need not necessarily to be removed and CNT bundles are not needed for each nanoelectronic signal as long as metallic CNTs can be detected and located. Upon detection and location, metallic CNTs can be configured to form global interconnects if not avoided. Their low resistivity helps to reduce signal propagation delay in global interconnects which are critical to nanoelectronic system performance.

Open CNTs are expected to be prevalent in a CNT array, as open CNT occurrence is proportional to the length of the CNT. A CNT with a single open can be largely included in a correct nanoelectronic design, upon detection and location of the single defect. A CNT with two (or more) opens is not fully utilizable. The segment between the two (extreme) opens are not accessible by any nano-addressing circuit, and components attached to that segment are not configurable. Upon detection and location of the extreme opens, the end segments of an open CNT can be included in a nano-circuit. Or, we can simply avoid open CNTs.

CNTs which are supposedly-parallel may cross over each other, resulting in different addresses for a CNT on two sides of a crossbar, and unexpected resistive contacts between CNTs. If not corrected by etching (34), such crossover CNTs can be taken as multi-thread cables and included in a correct nano-circuit. It is necessary to solve the following problem for nanoelectronic system configuration on a CNT crossbar nano-architecture.

Problem 3. *Detect and locate metallic, open and crossover CNTs in an CNT array, which are addressed on both ends by nano-addressing circuits.*

Such metallic, open, and crossover CNTs can be captured in a $n \times n$ resistance matrix $\mathbf{R}_{CNT}$, where each entry $R_{CNT}(i,j)$ gives the resistance of CNT between the i-th CNT end and the j-th CNT end on the opposite sides of an array of n CNTs (if $i \neq j$, $R_{CNT}(i,j)$ gives the resistance of a crossover CNT, otherwise, $R_{CNT}(i,i)$ gives the i-th CNT's resistance).
Method 3 solves Problem 3 by giving such a $n \times n$ resistance matrix $\mathbf{R}_{CNT}$. With this CNT resistance matrix $\mathbf{R}_{CNT}$, we avoid open CNTs, and consider only semiconductive CNTs, metallic CNTs, and crossover CNT bundles (as multi-thread cables) for the rest of the calibration (Methods 4 and 5 and 2).

Algorithm 3: Metallic, Open, Crossover CNT Detection and Location

Input: Array of n CNTs with nano-addressing circuits on both ends (Fig. 1)
Output: Resistance map $\mathbf{R}_{CNT}$ for metallic, open, crossover CNTs

1. Configure all CNFETs as shorts
2. For each i
3. For each j
4. Address the i-th CNT on one end of CNT
5. Address the j-th CNT on the other end of CNT
6. Measure resistance $R_{CNT}(i,j)$
7. If $i = j$ and $R_{CNT}(i,j) \approx \infty$
8. Open CNT (i,j)
9. If $i \neq j$ and $R_{CNT}(i,j) \ll \infty$
10. Crossover CNT (i,j)
11. If $R_{CNT}(i,j) \approx R_{metallic}$
12. Metallic CNT (i,j)
13. If $R_{CNT}(i,j) \approx R_{semiconductive}$
14. Semiconductive CNT (i,j)

4.3.2 Opens and Shorts in Programmable Vias

A CNT junction with electrically bistable molecules is a programmable via, which is supposedly reconfigured as a conductive via or open. A catastrophic defect at such a junction can be either (1) permanent open, or (2) permanent short. It is necessary to solve the following problem for nanoelectronic system configuration on a CNT crossbar nano-architecture.

Problem 4. *Detect and locate permanently open or short vias in a CNT crossbar nano-architecture.*

Method 4 solves Problem 4 by giving two $m \times n$ resistance maps $\mathbf{R}_{Pmin}$ and $\mathbf{R}_{Pmax}$, where each entry $R_{Pmin}(i,j)$ or $R_{Pmax}(i,j)$ gives the resistance of a L-shaped path which includes the i-th CNT segment on the top(bottom) of the CNT crossbar, the j-th CNT segment on the left(right) of the CNT crossbar, and a programmable via which is configured as conductive or open, respectively. Given non-open CNTs, these resistance matrices give a defect map for permanently open or short vias.

Algorithm 4: Permanently Open or Short Via Detection and Location

Input: Two layers of $m \times n$ CNT crossbar with nano-addressing interface on four sides
Output: Resistance maps $\mathbf{R}_{Pmin}$ and $\mathbf{R}_{Pmax}$ for permanently open or short vias

1. For each non-open CNT i
2. For each non-open CNT j
3. Address i-th CNT from top(down) of crossbar
4. Address j-th CNT from left(right) of crossbar
5. Program via $V(i,j)$ to conductive
6. Measure path resistance $R_{Pmin}(i,j)$
7. Program via $V(i,j)$ to insular
8. Measure path resistance $R_{Pmax}(i,j)$
9. If $R_{Pmin}(i,j) = R_{Pmax}(i,j) \approx \infty$
10. Permanently open via $V(i,j)$
11. If $R_{Pmin}(i,j) = R_{Pmax}(i,j) \approx R_{CNT}(i,i)$ or $R_{CNT}(j,j)$
12. Permanently short via $V(i,j)$

4.3.3 Opens and Shorts in CNFETs

A CNT junction with dielectric and redox active molecules is supposedly reconfigured as a FET. A catastrophic defect could lead to (1) short between source and drain (e.g., due to channel punchthrough, no intrinsic channel area, redox active molecules cannot release charge), (2) short between gate and source or drain (e.g., due to dielectric breakthrough), or (3) constant open gate (e.g., redox active molecules cannot hold charge). It is necessary to solve the following problem for nanoelectronic system configuration on a CNT crossbar nano-architecture.

Problem 5. *Detect and locate permanently open or short CNFETs in a CNT crossbar nano-architecture.*

Shorts between CNFET gate and source or drain can be detected in a method which is similar to Method 4 but without via programming. Method 5 finds permanent opens or shorts between the source and the drain of a CNFET by giving a $m \times n$ resistance matrix $\mathbf{R}_{CNFET}$. Upon detection and location, these catastrophic defects (metallic, open and crossover CNTs, permanently open or short vias and CNFETs) can be included in a correct nano-circuit. Nano-circuit physical design needs to be adaptive to the presence of these catastrophic defects, and will be different from die to die, based on the catastrophic defect maps ($\mathbf{R}_{CNT}$, $\mathbf{R}_{Pmin}$, $\mathbf{R}_{Pmax}$, and $\mathbf{R}_{CNFET}$) for each die.

Algorithm 5: Permanently Open or Short CNFET Detection and Location

Input: CNFETs in crossbar with nano-addressing interface, CNT resistance matrix $\mathbf{R}_{CNT}$
Output: Resistance map $\mathbf{R}_{CNFET}$ for permanently open or short CNFETs

1. Configure all CNFETs as shorts
2. For each non-open CNT i
3. For each non-open CNT j
4. Address the i-th CNT on both ends
5. Configure CNFET (i, j) as open
6. Measure resistance $R_{CNFET}(i, j)$
7. If $R_{CNFET}(i, j) \approx R_{CNT}(i, i) \ll \infty$
8. Short between CNFET (i, j) source-drain
9. If $R_{CNFET}(i, j) \approx R_{CNT}(i, i) \approx \infty$
10. Open between CNFET (i, j) source-drain
11. Configure CNFET (i, j) as short

4.4 Parametric Variation and Adaptive Design

Other than catastrophic defects, process variations are also critical to nanoelectronic system performance and reliability. Compared with catastrophic defects, process variations are more prevalent, and they are more difficult to detect since their effects are accumulated in affecting the underlying circuit. Adaptive or resilient nano-circuit design techniques are expected to achieve functionality and reliability in the presence of such process variations, besides online calibration and adaptive configuration as follows.

In adaptive configuration, each module of the circuit is configured with its test circuit. The test circuit can be as simple as additional interconnects which connect the inputs and the outputs of the module to some of the primary inputs and the primary outputs, respectively. In such cases, function and performance calibration is performed externally. Alternatively, self-test can be performed given the complexity of the test circuit. If the current configuration passes online function and performance verification, the auxiliary test circuit will be removed, and the current configuration of the module is committed. Otherwise, the same circuit module needs to be realized using other hardware resources on the reconfigurable platform.

5. Reliable, High Performance and Low Power Nano-Circuits

5.1 Nano-Circuit Design Challenges and Promising Techniques

As we have seen, the CNT crossbar nano-architecture provides regularity and manufacturability for high logic density implementations of all CMOS combinational logic families, including static logic, domino logic, pass-transistor logic, as well as latches, flip-flops, memory input address decoder and output sensing circuits such as differential sense amplifiers (Fig. 6 and Fig. 7).

However, nano-circuits in a CNT crossbar nano-architecture face a number of unique challenges. Nano-circuits must achieve reliability in the presence of prevalent defects and signif-

icant parametric variations, must achieve performance with highly resistive CNT interconnects, etc. We discuss nano-circuit design in a CNT crossbar nano-architecture in this section.
Nanoscale computing systems are expected to be subject to prevalent defects and significant process and environmental variations inevitably as a result of the uncertainty principle of quantum physics. E.g., the conductance of a CNT or a CNFET is very sensitive to chirality, diameter, etc. (38). Besides adaptive configuration, nanoscale computing systems need new computing models or circuit paradigms for reliability enhancement, performance improvement and power consumption reduction.
As technology scales, nanoelectronic computing systems are expected to be based on single electron devices (the average number of electrons in a transistor channel is approaching one for the current technologies). In quantum mechanics, the occurrence probability of an electron is the wavefunction given by the Schrödinger equation. How to extract a deterministic computation result from stochastic events such as electron occurrences is one of the fundamental problems that we are facing in designing nanometer scale computing systems. Traditional computation based on large devices can be modeled as redundancy and threshold based logic (which includes majority logic). In redundancy and threshold based logic, the error rate is given by a binomial distribution (the probability of observing m events in an environment of expecting an average of n independent events). As a result, a minimum signal-to-noise ratio is required with performance and power consumption implications. Finding a more efficient reliable computing model is of essential interest in nanoelectronic design.
Besides stochastic signal occurrence, signal propagation delay variability is another category of uncertainty in stochastic nanoscale systems. Nanoelectronic design needs to be adaptive to or resilient in the presence of signal propagation delay variations. Existing techniques (e.g., the Razor technology (2; 18) wherein a shadow flip-flop captures a delayed data signal for timing verification and correction) achieves only limited performance adaptivity, e.g., the circuity is adaptive to performance variations only within a given range. Asynchronous circuits have unlimited adaptivity to performance variations, and are ideal for high performance (enabling performance scaling in the presence of significant performance variations) and low power (being event-driven and clockless) nanoelectronic design (e.g., multi-core chips are expected to be increasingly self-timed, global-asynchronous-locally-synchronous, or totally asynchronous). However, existing asynchronous design techniques suffer in reliability in the presence of soft errors (e.g., glitches, coupling noises, radiation or cosmos ray strike induced random noises), which has limited their applications for decades.
A number of robust design techniques at multiple levels help to enhance reliability and reduce error rate of a nanoelectronic computing system. At the circuit level, differential signaling and complementary logic reduces parametric variation effects by exploiting spatial and temporal correlations (e.g., by correlating m and n for reduced error rate in a binomial distribution) (15; 31; 57). At a higher level, we believe that Error Detection/Correction Code (35) is the key to the stochastic signal occurrence problem in nanoscale systems (e.g., for lower required signal-to-noise ratios, which lead to high performance and low power), and needs to be applied more extensively at a variety of design hierarchy levels. Error correction coding has been applied widely in today's memories and wireless communication systems. Proposals also exist for applying (AN or residue) error detection/correction coding in arithmetic circuits (7).

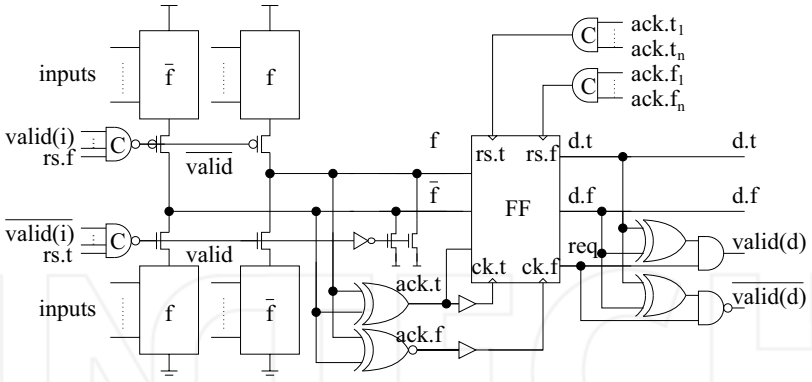

Fig. 11. A static logic based RDA (robust differential asynchronous) circuit.

5.2 Robust Asynchronous Circuits

A promising CNT nano-circuit paradigm is robust asynchronous circuits, which are formed by applying Error Detection/Correction Code to asynchronous circuit design for enhanced reliability in the presence of soft errors, while achieving delay insensitive nano-circuits (28).

For a 1-bit data signal, a robust differential asynchronous (RDA) communication channel includes three rails, two differential data signals $d.t$ and $d.f$, and a request signal req. The data and request signals are valid only if $(d.t, d.f, req) = (1,0,1)$ or $(0,1,1)$, while $(d.t, d.f, req) = (0,0,0)$ when the data and request signals return to zero. A validity check circuit detects the arrival of valid data and triggers the receiver flip-flop. The Hamming distance between any two legal codes for the data and request signals is at least two, instead of one in the dual-rail asynchronous signaling $(d.t, d.f) = (0,1), (1,0)$ or $(0,0)$. Differential acknowledgment signals $ack.t$ and $ack.f$ are also included.[2]

Fig. 11 gives a static logic based RDA circuit. [3] Two Muller C elements take the input validity signals $valid(i)$ and $\overline{valid(i)}$ (as well as the $rs.t$ and $rs.f$ signals coming from the downstream acknowledgment signals) and generate two differential validity signals $valid$ and $\overline{valid}$, which trigger the combinational logic computation. The Muller C elements have the maximum signal propagation delay for a rising(falling) output for all combinational logic circuits with the same number of inputs, due to the presence of the longest path of transistors to the power supply or the ground. Also the $valid$ and $\overline{valid}$ signals derive from and arrive later than the data signals. As a result, the $valid$ and $\overline{valid}$ signals always arrive later than the differential combinational logic outputs f and $\bar{f}$.

The later arriving validity signals filter out the internal glitches which come from combinational logic computation before the differential outputs f and $\bar{f}$ settle to their final values. Two NMOS transistors clamp the differential outputs f and the $\bar{f}$ to the ground until the $valid$ signal rises and the $\overline{valid}$ signal falls for noise immunity. The differential acknowledgment

[2] In general, multiple-bit data can be encoded in a variety of error detection codes (35). A single parity bit for n-bit data provides a Hamming distance of two which is immune to any single bit error, while an error detection code of a Hamming distance larger than k is immune to any k-bit error.

[3] Alternative implementations (in dynamic circuits, e.g., dual-rail domino, DCVSL, etc.) achieve different cost and reliability tradeoffs, and are potentially preferrable depending on the manufacturing technology and the environment, e.g., parametric variabilities, soft error rates, etc.

signals $ack.t$ and $ack.f$ derive from and always appear later than the differential data signals f and $\bar{f}$.
At the sender's end of the interconnect, for sequential elements, flip-flops are preferred over latches for reliability. A flip-flop is only vulnerable to noise when capturing the signal, while a latch is vulnerable to noise whenever it is transparent. The flip-flops send out differential data signals $d.t$ and $d.f$ (which come from the differential combinational logic outputs f and $\bar{f}$) as well as a request req signal (which comes from the acknowledgment signal $ack.t$). At the receiver end of the interconnect, a group of XOR and AND(NAND) gates verify the differential data and request signals, and generate two differential validity signals $valid(d)$ and $\overline{valid(d)}$. Any single bit soft error or common multiple bit soft errors injected to the interconnects or at the validity signals will halt the circuit.
Each sender flip-flop is triggered by two differential acknowledgment signals $ack.t$ and $ack.f$ as the differential clock signals $ck.t = 1$ and $ck.f = 0$, and is reset by two differential reset signals when $rs.t = 1$ and $rs.f = 0$. The differential reset signals come from the downstream differential acknowledgment signals $ack.t_i$ and $ack.f_i$ via the Muller C elements. They also generate the $valid$ and $\overline{valid}$ signals which trigger the combinational logic computation.
In the presence of multiple fanouts, multiple sets of differential acknowledgment signals will be sent back to the upstream stage. With the Muller C elements holding the input validity signals, the early arriving acknowledgment signals hold until the latest acknowledgment signal arrive from the fanouts. At that time the Muller C elements close the inputs to the combinational logic block and reset the flip-flop at the upstream stage, which brings all differential data and request signals $d.t$, $d.f$ and req as well as the acknowledgment signals $ack.t$ and $ack.f$ back to the ground, completing an asynchronous communication cycle.

5.3 Logic and Timing Correctness of RDA Circuits

An RDA circuit achieves logic and timing correctness in the presence of a single bit soft error or common multiple bit soft errors given the physical proximity of the circuit components.

Definition 3 (Single Bit Soft Error)**.** A single bit soft error *is a glitch or toggling caused by a single event upset as a result of an alpha particle or neutron strike from radioactive material or cosmos rays.*

Definition 4 (Common Multiple Bit Soft Error)**.** A common multiple bit soft error *is glitches or togglings of the same magnitude and polarity caused by common noises such as capacitive or inductive interconnect coupling, or spatially correlated transient parametric (e.g., supply voltage, temperature) variations (19; 39; 60), which have near identical effects on components at close physical proximity.*

Theorem 1 (Logic Correctness)**.** *An robust differential asynchronous circuit achieves logic correctness at the event of a single bit soft error or common multiple bit soft errors.*

Proof. An RDA circuit achieves logic correctness in the following cases.

1. A single bit soft error or a common multiple bit soft error at the input data signals leads to invalid data. The $valid(i)(\overline{valid(i)})$ signal will not rise(fall).
2. A single bit soft error or a common multiple bit soft error at the $valid(i)$ and $\overline{valid(i)}$ signals, at the Muller C elements computing the $valid$ and $\overline{valid}$ signals, or at the $valid$ and $\overline{valid}$ signals, leads to an early false or a late $valid(\overline{valid})$ signal. In this case, the differential structure for the $valid/\overline{valid}$ signals prevents any logic error. A false $valid$ signal turns off the keeper circuit, but can rise neither f nor $\bar{f}$, because the $\overline{valid}$ signal is still high. A false $\overline{valid}$ signal rises either f or $\bar{f}$, while the keeper circuit still clamps

both f and $\bar{f}$ to the ground. Only when both the $valid$ and $\overline{valid}$ signals arrive, the differential combinational logic computation is enabled.

3. A single bit soft error or a common multiple bit soft error at the differential combinational logic block or the differential data signals f and $\bar{f}$ will not raise the $ack.t$ signal nor lower the $ack.f$ signal.[4]
4. A single bit soft error or a common multiple bit soft error at the differential acknowledgment signals $ack.t$ and $ack.f$ does not trigger the flip-flop.
5. A single bit soft error or a common multiple bit soft error at the differential reset signals RS and $\overline{RS}$ does not reset the flip-flop.
6. A single bit soft error or a common multiple bit soft error at the differential data signals $d.t$ and $d.f$ and the request req signal leads to invalid data and does not generate a validity signal.

In summary, in order to make an RDA circuit to fail, the glitches must follow certain specific patterns, e.g., to reverse a "01" to a "10", which is highly unlikely to take place.

□

Theorem 2 (Timing Correctness)**.** *An robust differential asynchronous circuit achieves timing correctness for any delay variation given the physical proximity of the circuit components.*

Proof. Prevalent parametric (process, temperature, supply voltage) variations in nanoelectronic circuits lead to significant delay variations for the components in the circuit. Because such delay variations are spatially correlated (19; 39; 60), given the physical proximity of the circuit components, their delay variations are tightly correlated. Consequently, an RDA circuit achieves timing correctness in the following cases.

1. The input data signals $d.t$ and $d.f$ always arrive earlier than the differential validity signals $valid(d)$ and $\overline{valid(d)}$, which derive from the differential data and the request signals.
2. The $valid(\overline{valid})$ signal derives from the input validity signals $valid(i)$ and $\overline{valid(i)}$ through the Muller C elements.

   The rising(falling) delay of an Muller C element is the maximum of any combinational logic block of the same number of inputs with the longest serial transistor path to the power supply and the ground. Given the tight correlation of parametric variations for the transistors and the interconnects in the circuit, the $valid$ and $\overline{valid}$ signals arrive no early than the final combinational logic computation results f and $\bar{f}$. The $valid$ and $\overline{valid}$ signals enable the differential outputs f and $\bar{f}$ via the tri-state output structure and the keeper circuit, thus filtering out the glitches in combinational logic computation.
3. The differential acknowledgment signals $ack.t$ and $ack.f$ derive from and arrive no early than the differential data signals f and $\bar{f}$. They are further delayed (e.g., via buffers) such that at the rising edge of the flip-flop clock signal, the input data signals have settled to their final values, and the flip-flop captures the correct data f and $\bar{f}$. Consequently, no setup time constraint is required. The differential data signals f and $\bar{f}$ hold

[4] A glitch does not appear before valid data given the clamp NMOS transistors and fast data rise time in a static RDA circuit. A dynamic RDA circuit needs to enhance robustness for a soft error strike at a combinational logic output, e.g., by including weak keeper PMOS transistors, or delaying precharge PMOS transistors.

until the differential acknowledgment signals $ack.t$ and $ack.f$ reach the upstream stage, reset the upstream flip-flop, and lower the $valid(i)$ and $valid$ signals, which take much longer time than the hold time of the flip-flop. Consequently, no hold time constraint is required.

4. The differential acknowledgment signals $ack.t$ and $ack.f$ arrive after the combinational logic computation completes and the differential data signals f and $\bar{f}$ settle to their final values.
5. After all downstream stages send back acknowledgment signals, the flip-flop is reset, bringing the differential data $d.t$ and $d.f$ and the request req signals to the ground. The downstream stage acknowledgment signals are also brought back to the ground as a result. This completes a four-phase asynchronous communication cycle.

As a result, the proposed robust differential asynchronous circuit is delay insensitive, i.e., achieves correct timing (signal arrival time sequence) in the presence of delay variations, which is critical for nanoelectronic circuits. □

6. Experiments

6.1 Voltage-Controlled Nano-Addressing

In this section, we first verify the effectiveness of the proposed voltage-controlled nano-addressing circuit (Fig. 8) by running SPICE simulation based on the Stanford CNFET compact model (52).

In the proposed voltage-controlled nano-addressing circuit, each nanotube is gated by two N-type MOSFET-like CNFETs. These CNFETs are of $6.4nm$ gate width and $32nm$ channel length, as are described in the Stanford CNFET compact model. The two CNFETs in each nanotube are given a voltage drop of $V_{dd} = 1V$. The external address voltages are $V_{dda1} = V_{dda2} = 1V$, $V_{ssa1} = V_{ssa2} = 0$. As a result, the CNFETs have complementary gate voltages $V_{g1} + V_{g2} = 1V$. Fig. 12 gives the nanotube currents in the array with different gate voltage at the first address line. The nanotubes carry a significant current only with specific gate voltages, e.g., reaching $I_{out} = 5.064mA$ at gate voltage $V_{g1} = 0.495V$.

With 0 and $1V$ external voltages, Fig. 12 gives the currents for all the nanotubes in the array. With larger external voltages, Fig. 12 is extended to give the nanotube currents: any nanotube with a $V_{ga1} > 1V$ or $V_{ga2} < 0V$ gate voltage at the first address line carries zero current. Addressing resolution is given by the difference of addressing voltages between two adjacent nanotubes (since MOSFETs and MOSFET-like CNFETs are limited to a $< 60mV/decade$ inverse subthreshold slope). Adjusting the external address voltages minimizes any addressing inaccuracy due to manufacturing process and system runtime parametric variations.

6.2 Comparison of CNT Crossbar based and the Existing Nano-Architecture

Let us now compare nano-circuits implemented in the proposed CNT crossbar based nano-architecture and the existing nano-architectures. Considering DNA-guide self-assembly based nanoelectronic architectures such as NANA (42) and SOSA (43) target the far future, and FPNI (50) is very similar to CMOS technology by employing CMOS transistors and nanowires, I will compare RDG-CNFET based logic implementation with molecular diode and MOS transistor based logic implementation which is the mainstream nanoelectronic architecture in literature.[5]

[5] Comparing CNFET and CMOS-FET circuits gives approximately 5× performance improvement (16).

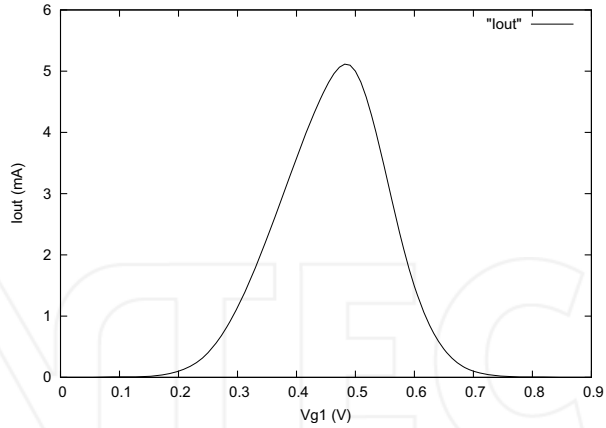

Fig. 12. Nanotube current I_{out} in mA for CNFET gate voltage V_{g1} in the first address line.

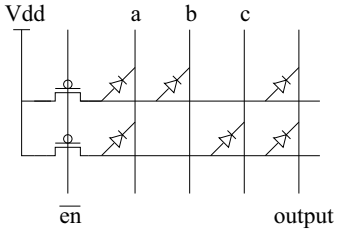

Fig. 13. A molecular diode/MOSFET based Boolean logic $a(b+c)$ implementation.

As an example of a combinational logic block, a Boolean logic function $a(b+c)$ is implemented based on RDG-CNFETs (Fig. 6) and by molecular diodes and peripheral CMOS transistors (Fig. 13). In the following experiments, SPICE simulation is conducted based on the latest Stanford compact CNFET model (52), a molecular device model from a latest publication (17), and the latest Predictive CMOS Technology Model (45).

The RDG-CNFETs are constructed based on an enhancement mode CNFET of $6.4nm$ gate width and $32nm$ channel length, as is described in the Stanford compact model (52). The bistable molecules at the front gate provide a resistance difference between $G\Omega$ and about 80Ω (6).[6] The redox active molecules at the back gate are cobalt phthalocyanine (CoPc) (17), which have been the basis of a NW-FET device with $1000\times$ conductance difference (17).

The molecular diodes are based on V-shaped amphiphilic [2]rotaxane 5^{4+} molecules (44), with saturation current $I_s = 36pA$, emission coefficient $N = 14.66$, and an on/off current ratio of 194.9. The CMOS transistors are modeled by $22nm$ Predictive Technology Models (45). To balance the current difference between molecular diodes and PMOS transistors, the PMOS transistors have a channel width/length ratio $W/L = 1/10$, while each molecular diode consists of $10,000$ V-shaped amphiphilic [2]rotaxane 5^{4+} molecules. As a result, the circuit has a current on the order of nA.

[6] The amorphous silicon based anti-fuse technology works with silicon based nanowires (6). Similar technologies are expected and assumed here for carbon nanotubes.

	Mo. Diode		RDG-CNFET	
abc	V_{out} (V)	P_{static} (W)	V_{out} (V)	P_{static} (W)
111	0.999	1.49n	1.000	0.25n
110	0.807	0.83μ	1.000	0.33n
101	0.807	0.83μ	1.000	0.32n
011	0.497	1.45μ	0.000	15.34p
000	0.265	1.47μ	0.000	41.32p

Table 1. Output voltage and static power consumption with different inputs of RDG-CNFET and molecular diode based Boolean logic $a(b+c)$ implementations.

Comparing the CNFET based and the molecular diode/CMOS based logic implementations, we have the following observations.

1. Area: The CNFET based logic implementation takes an area of $2 \times 6 = 12$ CNFETs and $2 \times 3 = 6$ vias, while molecular diodes and MOSFET based implementation takes an area of $2 \times 4 = 8$ molecular diodes and 2 MOSFETs (and two more MOSFETs if an inverter is included at each output to restore signal voltage swing). Considering CNFET based implementation is in a complementary logic, and the MOS transistors do not scale well, CNFET based implementation is expected to achieve superior logic density at a nanometer technology node.
2. Signal reliability: The CNFET based logic implementation achieves full voltage swing at the outputs, while in the diode logic circuit, the output swing depends on the inputs, and varies between $0.503V$ to $0.735V$ in the experiment (Table 1). Additional CMOS circuitry (e.g., an inverter) can be included at each output to restore full voltage swing, however, the reduced signal voltage swing in the diode logic circuit still implies compromised signal reliability.
3. Static power: The CNFET based logic implementation in CMOS logic achieves orders of magnitudes of less power consumption compared with molecular diodes and MOSFET based implementation for most input vectors (Table 1).
4. Performance: The CNFET based logic implementation achieves orders of magnitude of timing performance improvement compared with molecular diodes and MOSFET based implementation (Table 2).

In summary, CNFET based logic implementation achieves superior logic density, reliability, performance, and power consumption compared with molecular diodes and CMOS-FET based Boolean logic implementation.

6.3 Verification of RDA Circuits

In this section, we verify logic and timing correctness of robust differential asynchronous circuits by running HSPICE simulation based on $22nm$ Predictive Technology Models (45).
Fig. 14 gives signal waveforms for a perfect RDA circuit implementing Boolean function $f = \overline{ab}$ in CMOS static logic. Input signal a falls from logic 1 at $100ps$ to logic 0 at $200ps$, input signal b rises from logic 0 at $0ps$ to logic 1 at $50ps$. The validity signal $valid(\overline{valid})$ rises(falls) from logic 0(1) at $100ps$ to logic 1(0) at $200ps$. As a result, we observe that the late arrival of the $valid(\overline{valid})$ signals enable the combinational logic computation for f and $\bar{f}$, and filter out

	Mo. Diode		RDG-CNFET	
C_L (fF)	D_r (ns)	D_f (ns)	D_r (ns)	D_f (ns)
1	0.37	0.63	0.01	0.01
10	3.31	6.61	0.08	0.08
100	32.77	62.29	0.77	0.78

Table 2. Rising/falling signal propagation delays D_r/D_f (ns) (from a to $output$) for various load capacitance C_L (fF) of RDG-CNFET and molecular diode based Boolean logic $a(b+c)$ implementations. Input signal transition time varying from $1ps$ to $100ps$ leads to no considerable delay difference.

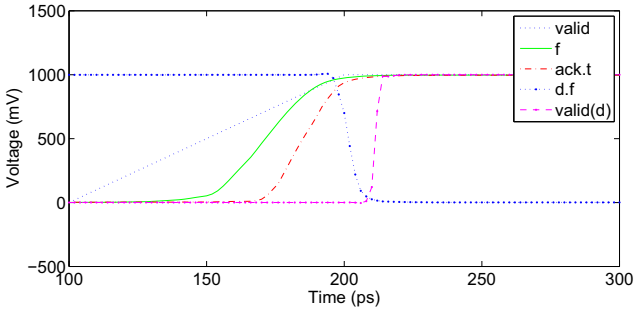

Fig. 14. Signal waveforms in a robust differential asynchronous circuit with no single bit soft error.

the internal glitches that would be observed at the f and $\bar{f}$ signals if the validity signals arrive early.

Fig. 15 gives signal waveforms for the same RDA circuit with the same input signals a and b, while the validity signal $valid$ is delayed by $50ps$ compared to the complementary validity signal $\overline{valid}$, representing an early false validity signal ($\overline{valid}$) or a late arriving validity signal ($valid$) due to an injected negative glitch at either the $\overline{valid}$ or the $valid$ signal. We observe that the differential data signals f and $\bar{f}$ are clamped to the ground until both validity signals settle to their final values $valid = 1$ and $\overline{valid} = 0$. As a result, no logic malfunction is present while all signals are delayed by $50ps$.

Fig. 16 gives signal waveforms for the RDA circuit with a triangle current (of $0.1mA$ peak current starting at $160ps$ ending at $180ps$) injected to the f signal. Comparing with Fig. 14, we observe that such a negative glitch does not lead to any logic error, instead, the arrivals of the $ack.t$ and $ack.f$ signals are postponed for about $40ps$, as well as all the downstream signals $d.t$, $d.f$, and $valid(d)$.

Fig. 17 gives signal waveforms for the RDA circuit with a triangle current (of $0.1mA$ peak current starting at $120ps$ ending at $140ps$) injected to the $ack.t$ signal. The glitch at the $ack.t$ signal does not trigger the flip-flop, and the subsequent signals $d.t$, $d.f$, $valid(d)$ and $\overline{valid(d)}$ are not affected.

Fig. 18 gives signal waveforms for the RDA circuit with two identical triangle currents (of $0.1mA$ peak current starting at $120ps$ ending at $140ps$) injected to both differential acknowl-

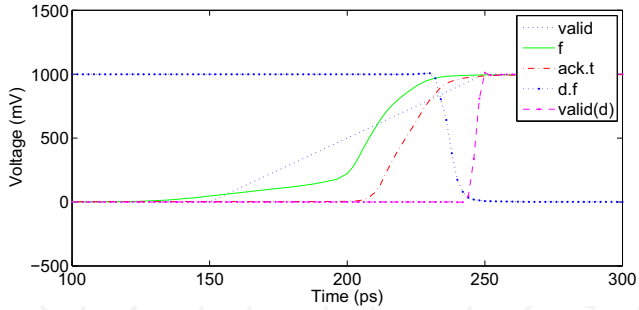

Fig. 15. Signal waveforms in a robust differential asynchronous circuit with an early false $\overline{valid}$ or a late arriving $valid$ signal.

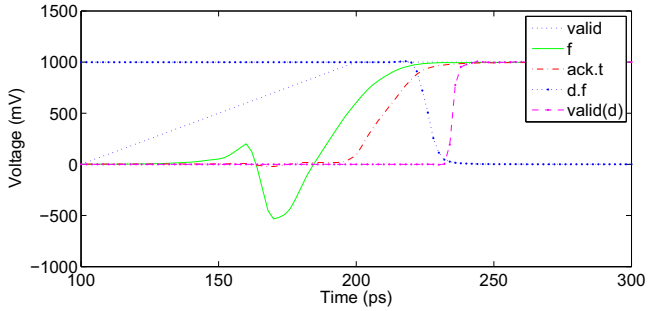

Fig. 16. Signal waveforms in a robust differential asynchronous circuit with a negative glitch injected at the f signal.

edgment signals $ack.t$ and $ack.f$. We observe that the double glitches do not trigger the flip-flop either, the subsequent signals $d.t$, $d.f$, $valid(d)$ and $\overline{valid(d)}$ are not affected.
From these experiments, we observe that the RDA circuit achieves correct logic and correct timing (signal arrival time sequence) at the event of a single bit soft error or common multiple bit soft errors, by temporarily halting the circuit operation until the valid data re-appear.

7. Conclusion

In this chapter, we have studied the first purely CNT and CNFET based nano-architecture, which is based on (1) a novel reconfigurable double gate carbon nanotube field effect transistor (RDG-CNFET) device, (2) a multi-layer CNT crossbar structure with sandwiched via-forming and gate-forming molecules, and (3) a novel voltage-controlled nano-addressing circuit not requiring precise layout design, enabling manufacture of nanoelectronic systems in all existing CMOS circuit design styles.
A complete methodology of adaptive configuration of nanoelectronic systems based on the CNT crossbar nano-architecture is also presented, including (1) an adaptive nano-addressing method for the voltage-controlled nano-addressing circuit, (2) an adaptive RDG-CNFET gate matching method, and (3) a set of catastrophic defect mapping methods, which are specific (to

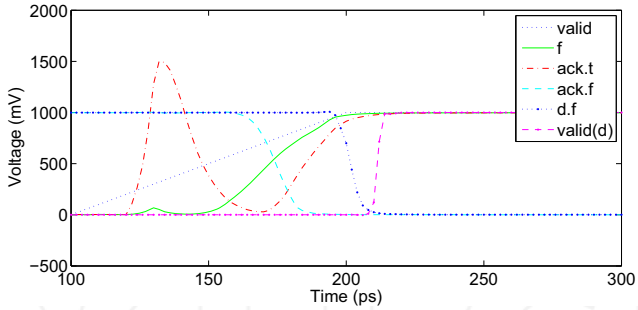

Fig. 17. Signal waveforms in a robust differential asynchronous circuit with a positive glitch injected at the *ack.t* signal.

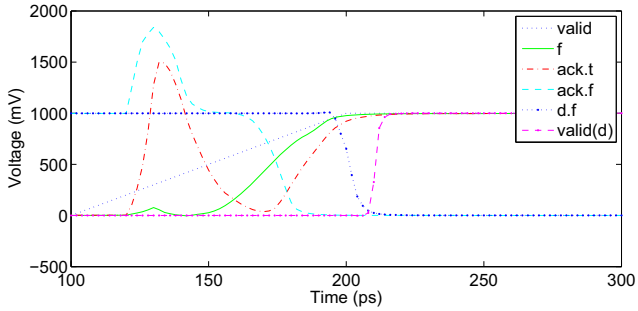

Fig. 18. Signal waveforms in a robust differential asynchronous circuit with positive glitches injected at both differential acknowledgment signals *ack.t* and *ack.f*.

the CNT crossbar nano-architecture), complete (in detecting and locating all possible catastrophic defects in the CNT crossbar nano-architecture), deterministic (with no probabilistic computation), and efficient (test paths are rows or columns of CNT, or L-shaped CNT paths, CNT open/short defect detection is separated with via/CNFET open/short defect detection, runtime is linear to the number of defect sites). This is significant improvement compared with the previous techniques (which either detects only a single defect (10), or is generic, abstract, probabilistic, and highly complex (8; 33; 58)).

We have also examined some of the design challenges and promising techniques for CNT and CNFET based nano-circuits. We identify significant parametric variation effects on logic correctness and timing correctness, and propose robust (differential) asynchronous circuits by applying Error Detection Code to asynchronous circuit design for noise immune and delay insensitive nano-circuits.

SPICE simulation based on compact CNFET and molecular device models demonstrates superior logic density, reliability, performance and power consumption of the proposed RDG-CNFET based nanoelectronic architecture compared with previously published nanoelectronic architectures, e.g., of a hybrid nano-CMOS technology including molecular diodes and MOSFETs. Furthermore, theoretical analysis and SPICE simulation based on 22nm Predictive Technology Models show that RDA circuits achieve much enhanced reliability in logic correct-

ness in the presence of a single bit soft error or common multiple bit soft errors, and timing correctness in the presence of parametric variations given the physical proximity of the circuit components.

While nanotechnology development has not enabled fabrication of such a system, this chapter has demonstrated the prospected manufacturability, reliability, and performance of a purely carbon nanotube and carbon nanotube transistor based nanoelectronic system. These nanoelectronic system design techniques are expected to be further developed along with nanoscale device fabrication and integration techniques which are critical to achieve and to improve the proposed nanoelectronic architecture in several aspects, including: (1) search of electrically bistable molecules of repeated reconfigurability and low contact resistance with carbon nanotubes, (2) development of etching processes for electrically bistable and redox active molecules with carbon nanotubes as masks, and (3) manufacture of nanoscale devices of superior subthreshold slope for enhanced nano-addressing resolution, and ultra high performance low power nanoelectronic systems.

8. References

[1] I. Amlani, N. Pimparkar, K. Nordquist, D. Lim, S. Clavijo, Z. Qian and R. Emrick, "Automated Removal of Metallic Carbon Nanotubes in a Nanotube Ensemble by Electrical Breakdown," *Proc. IEEE Conference on Nanotechnology*, 2008, pp. 239-242.

[2] T. Austin, V. Bertacco, D. Blaauw and T. Mudge, "Opportunities and Challenges for Better Than Worst-Case Design," *Asian South Pacific Design Automation Conference*, 2005.

[3] A. Bachtold, P. Hadley, T. Nakanishi and C. Dekker, "Logic Circuits with Carbon Nanotube Transistors," *Science*, 2001, 294(5545), pp. 1317-1320.

[4] P. Beckett, "A Fine-Grained Reconfigurable Logic Array Based on Double Gate Transistors," *International Conference on Field-Programmable Technology*, 2002, pp. 260-267.

[5] L. Benini and G. De Micheli, "Networks on chips: A new paradigm for component-based MPSoC design," *Proc. MPSoC.*, 2004.

[6] J. Birkner, A. Chan, H. T. Chua, A. Chao, K. Gordon, B. Kleinman, P. Kolze and R. Wong, "A Very High-Speed Field Programmable Gate Array Using Metal-To-Metal Anti-Fuse Programmable Elements," *New Hardware Product Introduction at Custom Integrated Circuits Conference,* 1991.

[7] T. J. Brosnan and N. R. Strader II, "Modular Error Detection for Bit-Serial Multiplication," *IEEE Trans. on Computers*, 37(9), 1988, pp. 1043-1052.

[8] J. G. Brown and R. D. Blanton, "CAEN-BIST: Testing the NanoFabric," *Proc. International Test Conference*, 2004, pp. 462-471.

[9] Z. Chen, J. Appenzeller, Y.-M. Lin, J. Sippel-Oakley, A. G. Rinzler, J. Tang, S. J. Wind, P. M. Solomon and P. Avouris, "An Integrated Logic Circuit Assembled on a Single Carbon Nanotube," *Science*, 2006, 311(5768), pp. 1735.

[10] B. Culbertson, R. Amerson, R. Carter, P. Kuekes and G. Snider, "Defect Tolerance on the Teramac Custom Computer," *Proc. Symposium on FPGA's for Custom Computing Machines (FCCM)*, 2000, pp. 185-192.

[11] A.DeHon, "Array-Based Architecture for FET-Based, Nanoscale Electronics," *IEEE Trans. on Nanotechnology*, 2(1), pp. 23-32, 2003.

[12] A. DeHon, P. Lincoln and J. E. Savage, "Stochastic Assembly of Sublithographic Nanoscale Interface," *IEEE Trans. Nanotechnology*, 2(3), pp. 165-174, 2003.

[13] A. DeHon and M. J. Wilson, "Nanowire-Based Sublithographic Programmable Logic Arrays," *Proc. FPGA*, 2004, pp. 123-132.

[14] C. Dwyer, L. Vicci, J. Poulton, D. Erie, R. Superfine, S. Washburn and R. M. Taylor, "The design of DNA self-assembled computing circuitry," *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, 12(11), pp. 1214-1220, Nov. 2004.
[15] D. J. Deleganes, M. Barany, G. Geannopoulos, K. Kreitzer, M. Morrise, D. Milliron, A. P. Singh and S. Wijeratne, "Low-Voltage Swing Logic Circuits for a Pentium 4 Processor Integer Core," *IEEE J. of Solid-State Circuits*, 40(1), pp. 36-43, 2005.
[16] J. Deng, and H.-S. P. Wong, "A Compact SPICE Model for Carbon Nanotube Field Effect Transistors Including Non-Idealities and Its Application âĂŤ Part II: Full Device Model and Circuits Performance Benchmarking," *IEEE Trans. Electron Devices*, 2007.
[17] X. Duan, Y. Huang and C. M. Lieber, "Nonvolatile Memory and Programmable Logic from Molecule-Gated Nanowires," *Nano Letters*, 2(5), pp. 487-490, 2002.
[18] D. Ernst, N. S. Kim, S. Das, S. Lee, D. Blaauw, T. Austin, T. Mudge and K. Flautner, "Razor: Circuit-Level Correction of Timing Errors for Low-Power Operation," *IEEE MICRO special issue on Top Picks From Microarchitecture Conferences of 2004*, 2005.
[19] P. Friedberg, Y. Cao, J. Cain, R. Wang, J. Rabaey, and C. Spanos, "Modeling Within-Die Spatial Correlation Effects for Process-Design Co-Optimization," *Proc. International Symposium on Quality Electronic Design*, pp. 516-521, 2005.
[20] S. C. Goldstein and M. Budiu, "NanoFabrics: Spatial Computing Using Molecular Electronics," *Proc. International Symposium on Computer Architecture*, 2001, pp. 178-191.
[21] B. Gojman, E. Rachlin, and J. E. Savage, "Evaluation of Design Strategies for Stochastically Assembled Nanoarray Memories," *Journal of Emerging Technologies,* 1(2), 2005, pp. 73-108.
[22] J. R. Heath and M. A. Ratner, "Molecular Electronics," *Physics Today*, 56(5), pp. 43-49, 2003.
[23] S. J. Kang, C. Kocabas, T. Ozel, M. Shim, N. Pimparkar, M. A. Alam, S. V. Rotkin and J. A. Rogers, "High-Performance Electronics Using Dense, Perfectly Aligned Arrays of Single-Walled Carbon Nanotubes," *Nature Nanotechnology*, Vol. 2, pp. 230-236, April 2007.
[24] B. S. Landman and R. L. Russo, "On a Pin Versus Block Relationship for Partitions of Logic Graphs," *IEEE Trans. on Computers*, C-20, pp. 1469-1479, 1971.
[25] J. Liu, I. O'Connor, D. Navarro and F. Gaffiot, "Design of a Novel CNTFET-Based Reconfigurable Logic Gate," *Proc. ISVLSI*, 2007, pp. 285-290.
[26] B. Liu, "Reconfigurable Double Gate Carbon Nanotube Field Effect Transistor Based Nanoelectronic Architecture," *Proc. Asia and South Pacific Design Automation Conference*, 2009.
[27] B. Liu, "Adaptive Voltage Controlled Nanoelectronic Addressing for Yield, Accuracy and Resolution," *Proc. International Symposium on Quality Electronic Design*, 2009.
[28] B. Liu, "Robust Differential Asynchronous Nanoelectronic Circuits," *Proc. International Symposium on Quality Electronic Design*, 2009.
[29] B. Liu, "Defect Mapping and Adaptive Configuration of Nanoelectronic Circuits Based on a CNT Crossbar Nano-Architecture," *Workshop on Nano, Molecular, and Quantum Communications (NanoCom)*, 2009.
[30] International Technology Roadmap for Semiconductors, `http://www.itrs.net/`.
[31] A. Maheshwari and W. Burleson, "Differential Current-Sensing for On-Chip Interconnects," *IEEE Trans. on VLSI Systems*, 12(12), 2004, pp. 1321-1329.
[32] P. L. McEuen, M. S. Fuhrer and P. Hongkun, "Single-walled Carbon Nanotube Electronics," *IEEE Trans. Nanotechnology*, 1(1), pp. 78-85, 2002.
[33] M. Mishra and S. C. Goldstein, "Defect Tolerance at the End of the Roadmap," *Proc. International Test Conference*, 2003, pp. 1201-1211.

[34] S. Mitra, N. Patil and J. Zhang, "Imperfection-Immune Carbon Nanotube VLSI Logic Circuits," *Foundations of NANO*, 2008.
[35] T. K. Moon, *Error Correction Coding: Mathematical Methods and Algorithms*, Wiley-Interscience, 2005.
[36] H. Naeimi and A. DeHon, "A Greedy Algorithm for Tolerating Defective Crosspoints in NanoPLA Design," *Proc. Intl. Conf. on Field-Programmable Technology*, 2004, pp. 49-56.
[37] P. Nguyen, H. T. Ng, T. Yamada, M. K. Smith, J. Li, J. Han and M. Meyyappan, "Direct Integration of Metal Oxide Nanowire in Vertical Field-Effect Transistor," *Nano Letters*, 2004, 4(4), pp. 651-657.
[38] A. Nieuwoudt and Y. Massoud, "Assessing the Implications of Process Variations on Future Carbon Nanotube Bundle Interconnect Solutions," *Proc. Intl. Symp. on Quality Electronic Design*, 2007, pp. 119-126.
[39] M. Orshansky, L. Milor, P. Chen, K. Keutzer, C. Hu, "Impact of spatial intrachip gate length variability on the performance of high-speed digital circuits," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, 2002, pp. 544-553.
[40] S. S. P. Parkin, "Spintronics Materials and Devices: Past, Present and Future," *IEEE International Electron Devices Meeting (IEDM) Technical Digest*, pp. 903-906, 2004.
[41] N. Patil, J. Deng, A. Lin, H.-S. Philip Wong and S. Mitra, "Design Methods for Misaligned and Mispositioned Carbon-Nanotube Immune Circuits," *IEEE Tran. on CAD*, 2008, 27(10), pp. 1725-1736.
[42] J. P. Patwardhan, C. Dwyer, A. R. Lebeck and D. J. Sorin, "NANA: A Nano-Scale Active Network Architecture," *ACM Journal on Emerging Technologies in Computing Systems,* 2(1), pp. 1-30, 2006.
[43] J. P. Patwardhan, V. Johri, C. Dwyer and A. R. Lebeck, "A Defect Tolerant Self-Organizing Nanoscale SIMD Architecture," *International Conference on Architecture Support for Programming Languages and Operating Systems,* 2006, pp. 241-251.
[44] A. R. Pease, J. O. Jeppesen, J. F. Stoddart, Y. Luo, C. P. Collier and J. R. Heath, "Switching Devices Based on Interlocked Molecules," *Acc. Chem. Res.*, 34, pp. 433-444, 2001.
[45] Predictive Technology Model, `http://www.eas.asu.edu/~ptm/`.
[46] A. Raychowdhury and K. Roy, "Carbon Nanotube Electronics: Design of High Performance and Low Power Digital Circuits," *IEEE Trans. on Circuits and Systems - I: Fundamental Theory and Applications,* 54(11), pp. 2391-1401, 2007.
[47] G. S. Rose, A. C. Cabe, N. Gergel-Hackett, N. Majumdar, M. R. Stan, J. C. Bean, L. R. Harriott, Y. Yao and J. M. Tour, "Design Approaches for Hybrid CMOS/Molecular Memory Based on Experimental Device Data," *Proc. Great Lakes Symposium on VLSI*, 2006, pp. 2-7.
[48] J. E. Savage, E. Rachlin, A. DeHon, C. M. Lieber and Y. Wu, "Radial Addressing of Nanowires," *ACM Journal of Emerging Technologies in Computing Systems,* 2(2), pp. 129-154. 2006.
[49] M. S. Schmidt, T. Nielsen, D. N. Madsen, A. Kristensen and P. BÃÿggild, "Nano-Scale Silicon structures by Using Carbon Nanotubes as Reactive Ion Masks," *Nanotechnology*, 16, pp. 750-753, 2005.
[50] G. S. Snider and R. S. Williams, "Nano/CMOS Architectures Using a Field-Programmable Nanowire Interconnect," *Nanotechnology*, 18(3), 2007.
[51] M. R. Stan, P. D. Franzon, S. C. Goldstein, J. C. Lach and M. M. Ziegler, "Molecular Electronics: From Devices and Interconnect to Circuits and Architecture," *Proc. of the IEEE*, 91(11), pp. 1940-1957, 2003.
[52] Stanford CNFET Model, `http://nano.stanford.edu/models.php`.

[53] R. Sordan, K. Balasubramanian, M. Burghard and K. Kern, "Exclusive-OR Gate with a Single Carbon Nanotube," *Appl. Phys. Lett.*, 2006, 88.
[54] D. B. Strukov and K. K. Likharev, "CMOL FPGA: A Reconfigurable Architecture for Hybrid Digital Circuits with Two-Terminal Nanodevices," *Nanotechnology*, 16(6), pp. 888-900, 2005.
[55] J. P. Sun, G. I. Haddad, P. Mazumder and J. N. Schulman, "Resonant Tunneling Diodes: Models and Properties," *Proc. of the IEEE*, 86(4), 1998, pp. 641-660.
[56] Y.-C. Tseng, P. Xuan, A. Javey, R. Malloy, Q. Wang, J. Bokor and H. Dai, "Monolithic Integration of Carbon Nanotube Devices with Silicon MOS Technology," *Nano Letters*, 4(1), pp. 123-127, 2004.
[57] N. Tzartzanis and W. W. Walker, "Differential Current-Mode Sensing for Efficient On-Chip Global Signaling," *IEEE J. of Solid-State Circuits*, 40(11), 2005, pp. 2141-2147.
[58] Z. Wang and K. Chakrabarty, "Using Built-In Self-Test and Adaptive Recovery for Defect Tolerance in Molecular Electronics-Based NanoFabrics," *Proc. International Test Conference*, 2005.
[59] R. S. Williams and P. J. Kuekes, "Demultiplexer for a Molecular Wire Crossbar Network," *US Patent Number 6,256,767*, 2001.
[60] J. Xiong, V. Zolotov, and L. He, "Robust Extraction of Spatial Correlation," *Proc. International Symposium on Physical Design*, 2006, pp. 2-9.
[61] J. Zhang, N. P. Patil, A. Hazeghi and S. Mitra, "Carbon Nanotube Circuits in the Presence of Carbon Nanotube Density Variations," *Proc. Design Automation Conference*, 2009.
[62] J. Zhang, N. P. Patil and S. Mitra, "Design Guidelines for Metallic-Carbon-Nanotube-Tolerant Digital Logic Circuits, *Proc. Design Automation and Test in Europe*, 2008, pp. 1009-1014.
[63] W. Zhang, N. K. Jha and L. Shang, "NATURE: A Hybrid Nanotube/CMOS Dynamically Reconfigurable Architecture," *Proc. Design Automation Conference*, 2006, pp. 711-716.
[64] G. Zhang, P. Qi, X. Wang, Y. Lu, X. Li, R. Tu, S. Bangsaruntip, D. Mann, L. Zhang and H. Dai, "Selective Etching of Metallic Carbon Nanotubes by Gas-Phase Reaction," *Science*, 314, 2006, pp. 974-977.

VLSI

Edited by Zhongfeng Wang

ISBN 978-953-307-049-0
Hard cover, 456 pages
Publisher InTech
Published online 01, February, 2010
Published in print edition February, 2010

The process of Integrated Circuits (IC) started its era of VLSI (Very Large Scale Integration) in 1970's when thousands of transistors were integrated into one single chip. Nowadays we are able to integrate more than a billion transistors on a single chip. However, the term "VLSI" is still being used, though there was some effort to coin a new term ULSI (Ultra-Large Scale Integration) for fine distinctions many years ago. VLSI technology has brought tremendous benefits to our everyday life since its occurrence. VLSI circuits are used everywhere, real applications include microprocessors in a personal computer or workstation, chips in a graphic card, digital camera or camcorder, chips in a cell phone or a portable computing device, and embedded processors in an automobile, et al. VLSI covers many phases of design and fabrication of integrated circuits. For a commercial chip design, it involves system definition, VLSI architecture design and optimization, RTL (register transfer language) coding, (pre- and post-synthesis) simulation and verification, synthesis, place and route, timing analyses and timing closure, and multi-step semiconductor device fabrication including wafer processing, die preparation, IC packaging and testing, et al. As the process technology scales down, hundreds or even thousands of millions of transistors are integrated into one single chip. Hence, more and more complicated systems can be integrated into a single chip, the so-called System-on-chip (SoC), which brings to VLSI engineers ever increasingly challenges to master techniques in various phases of VLSI design. For modern SoC design, practical applications are usually speed hungry. For instance, Ethernet standard has evolved from 10Mbps to 10Gbps. Now the specification for 100Mbps Ethernet is on the way. On the other hand, with the popularity of wireless and portable computing devices, low power consumption has become extremely critical. To meet these contradicting requirements, VLSI designers have to perform optimizations at all levels of design. This book is intended to cover a wide range of VLSI design topics. The book can be roughly partitioned into four parts. Part I is mainly focused on algorithmic level and architectural level VLSI design and optimization for image and video signal processing systems. Part II addresses VLSI design optimizations for cryptography and error correction coding. Part III discusses general SoC design techniques as well as other application-specific VLSI design optimizations. The last part will cover generic nano-scale circuit-level design techniques.

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

INTECH