Molecular Electronics: Devices, Systems and Tools for Gigagate, Gigabit Chips

Michael Butts
Cadence Design Systems, Inc.
13221 SW 68th Pkwy
Portland, OR 97223
⟨mbutts@cadence.com⟩

André DeHon
Dept. of CS, 256-80
California Institute of Technology
Pasadena, CA 91125
⟨andre@acm.org⟩

Seth Copen Goldstein
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
⟨seth@cs.cmu.edu⟩

Abstract

New electronics technologies are emerging which may carry us beyond the limits of lithographic processing down to molecular-scale feature sizes. Devices and interconnects can be made from a variety of molecules and materials including bistable and switchable organic molecules, carbon nanotubes, and, single-crystal semiconductor nanowires. They can be self-assembled into organized structures and attached onto lithographic substrates. This tutorial reviews emerging molecular-scale electronics technology for CAD and system designers and highlights where ICCAD research can help support this technology.

1. Introduction

Researchers have reported a number of interesting developments in the field of molecular-scale electronics during the past two years, prompting the AAAS journal *Science* to name Molecular Electronics as its "Breakthrough of the Year" for 2001. With these developments and an accelerated rate of progress, interesting molecular-scale circuits are moving from fantasy into reality. This decade it may become feasible to self assemble digital systems with billions of components. The impact on electronics, and ICCAD tools in particular, will be profound.

Molecular-scale devices actually operating today include: FETs, junction transistors, diodes, and, molecular and mechanical switches. Logic gates with voltage gain have been built, and many techniques have been demonstrated to assemble nanometer wide wires into large arrays. Programmable and non-volatile devices which hold their state in a few molecules or in square nanometers of material have been demonstrated. Researchers project densities of 10^{11} to 10^{12} devices/cm^2. (In comparison, the 2001 ITRS roadmap [1] projects 3×10^9 transistors/cm^2 for 2016 for ASICs and MPUs.)

Building systems of this size requires a completely new set of approaches to manufacturing and fabrication which will have a significant impact on circuits and architectures. These approaches are characterized by bottom-up molecular-scale fabrication which avoid the needs for lithographic patterning in order to create the nanometer scale dimensions for these devices and assemblies. We can expect batch chemical and physical processes which will be able to form semi-regular arrays with 1% to 5% defect rates. Interesting circuit functionality can be realized by a combination of statistical design and post-fabrication programming to avoid defects and define desired computations.

Cheap gigagate, gigabit, gigaHertz non-volatile FPGAs and RAMs, built with molecular-scale electronic arrays on microelectronic silicon substrates, may be ready for commercial production by 2010.

0-7803-7607-2/02/$17.00

In the next section we survey some of the promising, emerging device technologies. Section 3 surveys some fabrication techniques and their implications on the kinds of systems we can build. Next, in Section 4, we review proposals for designing complete electronic systems. Defect and fault tolerance, discussed in Section 5, will be major issues for any technology at this feature size where single molecules and countable electrons are used to form devices and hold state. In Section 6 we highlight some of the new challenges for CAD created by bottom-up, nanoscale technologies.

2. Materials and Devices

2.1. Robust Digital Abstraction

Over the past several decades, we have developed a series of abstractions and properties that, when met, make it easy to assemble and reason about robust digital systems, including: switching, isolation, noise margins, restoration, and state storage. Isolation allows active devices to be combined in large circuits and define a single, intended direction of signal flow. Signal restoration makes output logic levels match or exceed input logic levels, assuring that it is possible to cascade an arbitrary number of devices in series and for circuits to operate correctly with feedback cycles. Good noise margins reject noise in the system from crosstalk and voltage drops across dissapitive media. Gain in circuits is needed both for signal restoration and to support fanout.

We have successfully transitioned from vacuum tubes, to diodes, to bipolar transistors, and to MOS transitors as basic logic devices by maintaining these features and transporting our experience from one form of circuit technology to the next. Today's digital logic, memory and hardware system architectures can be built on any implementation technology that satisfies these requirements, including molecular-scale circuits. Molecular electronics, however, may require new ways of thinking about these properties. As we will describe below, the molecular devices, contacts, and wires often have far different characteristics than traditional devices. In many cases, the characteristics are currently inferior, but these weakness do not appear fundamental. For example, none of the current molecular devices have all of the above properties, but rather a conglomeration of molecular devices can be assembled to satisfy them.

2.2. Carbon Nanotubes

In 1991 Sumio Iijima discovered a tubular variant of the C_{60} "buckyball" carbon molecule [24]. A carbon nanotube (NT) is a molecular sheet a single atom thick, which has wrapped around into a tube, as little as one nanometer wide and, so far, up to millimeters long. Since it's a single molecule, nanotubes are extremely strong and flexible.

Depending on their lattice geometry, NTs behave as metals or semiconductors, acting as good electrical conductors. So far there is no way to synthesize a pure batch of just one type [35]. Some

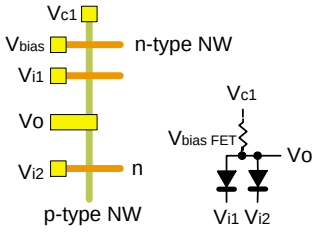


Figure 1: CNW-diode AND gate

success in selecting NTs by voltage breakdown of undesired types has been reported [9]. Carbon nanotube technology is very new. NTs may become more predictable and manageable in time.

2.3. Single-crystal Nanowires

Silicon may be the best understood and most manageable of all solids. Single-crystal silicon nanowires (NWs) have been fabricated in bulk by laser-assisted catalytic growth, with diameters of 6 to 20 nm and lengths ranging from 1 to 30 microns [25]. Germanium, gold, gallium phosphide, gallium nitride and indium phosphide NWs have also been made [22, 31].

Evaluating nanoscale wires is difficult because the wire/contact/device system is highly intertwined. In the best case, the wire behaves like a 1-D quantum wire, *i.e.*, it transports electrons ballistically, and has a length-independent resistance in multiples of h/e^2 [26]. A perfectly conducting nanotube has a minimal resistance of $h/4e^2$ or a resistance of approximately 6 KΩ; which agrees with experiment [36]. Capacitance continues to scale with length.

2.4. Transistors, Diodes, and NDRs

Nanowires and nanotubes can be used as more than just interconnect wires, they can also be used as active devices. Metallic wires can be grown with molecules which act as two-terminal devices embedded in them [33]. Controlled doping of silicon and gallium nitride NWs with phosphorus and boron has been successful, producing p-type and n-type semiconducting nanowires [10]. Carbon nanotubes can also act as diodes or transistors [21].

Two semiconductor NWs, one p-type and one n-type, form a junction diode at their crossing. Three NWs with two crossings form a bipolar junction transistor [11]. Small working NW diode arrays have been made, with 85% to 95% yield, showing independent operation. Turn-on voltages of 1V are observed, and logic gates may be made with crossed nanowire diodes (CNW-diodes) [21] (See Figure 1).

CNW-diodes can be made with 5V turn-on by increasing the oxide thickness at the junctions. This oxide can be grown by passing high current through a low turn-on diode in air, oxidizing the junction by joule heating [21]. This device can be used as a one-time-programmable crosspoint for ROM or logic arrays, or it can be used as an FET.

CNW-diodes act as FETs in their non-conducting region. A p-channel crossed nanowire FET (CNW-FET) has a p-type single-crystal silicon NW channel and n-type single-crystal gallium nitride NW gates. The resulting NOR gate (See Figure 2) has a voltage gain of 5 at room temperature [21], and was the first reported nanoscale logic gate with gain.

Molecular resonant tunneling diodes, often called negative differential resistors (NDRs), have also been realized [6]. These devices are characterized by a region of negative resistance in their

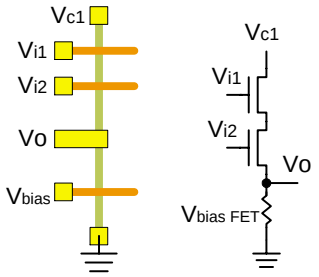


Figure 2: CNW-FET NOR gate

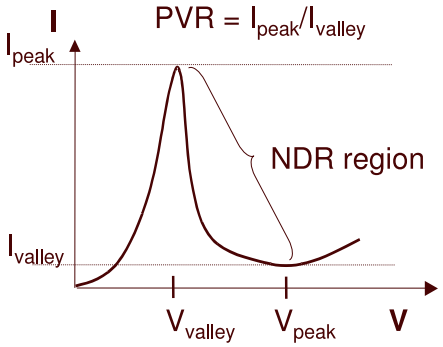


Figure 3: I-V curve for a "typical NDR device

IV-curve (See Figure 3.) Devices with useful Peak-to-valley ratio have been measured at room temperature [7]. These molecules can be used to as the basis of logic families [17, 14] or as the core of a molecular latch which also provides signal restoration and I/O-isolation [20].

2.5. Programmable Switches

Molecular electronics has the potential to not only reduce the scale of systems, but also to introduce novel devices with features not found in any silicon based systems. An example are programmable switches which hold their own state and can be programmed using the signal wires, yet take up no more area than a wiring crossing. In conventional silicon systems, a programmable crosspoint costs 40–100× the area of a wire crossing or via. Consequently, these devices reduce the overhead (in both device area and interconnect area) for programmable circuits.

Molecular Switches Organic molecules exist which have two mechanically distinct parts, such as a ring and a rod or interlocking rings. Applying a programming voltage across the molecule adds or subtracts an electron (oxidation-reduction), shifting the ring and changing the molecule's conductivity. It functions as a non-volatile programmable molecular switch. Catenane is one such molecule which opens at 2 volts, closes at 2 volts, is read at 0.1 volt, and has been cycled open and closed many times [8]. Used between a metal wire and an n-type silicon nanowire, the junction acts as a programmable diode, making an addressable memory array. So far its conductance only varies by 4× between states, so it may be useful for RAM but not for logic. A nitroamine [6] showed 1000× conductance variation between states.

Both these molecular switches and the mechanical switches below have no gain, so signals must be restored between switch arrays by devices with gain, such as CNW-FETs or NDR-based latches.

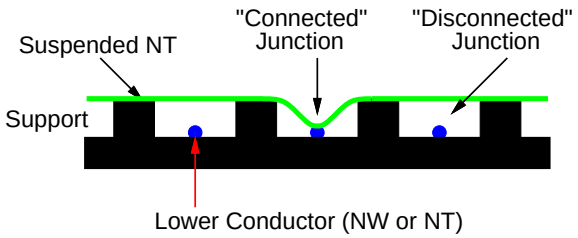


Figure 4: Suspended Nanotube Switched Connection

Mechanical Switches Carbon nanotubes arranged in a crossbar, with the upper half suspended so they are separated from the lower half, will work as a programmable array of non-volatile bistable mechanical switches (See Figure 4). Applying enough voltage across a pair of NTs attracts them together, then van der Waals forces keep them in contact, closing a circuit between the NTs. Applying an opposite voltage drives them apart again. This operation was simulated and a single working crosspoint was made [34]. Diodes at the crosspoints are needed for unique addressing without 'back door' paths. Using semiconductor NTs or NWs for the lower wires may make diodes at the contact points [34].

3. Fabrication

Perhaps the largest difference between molecular electronics and traditional VLSI is in the methods of fabrication. Molecular electronics is based on bottom-up manufacturing as opposed to the traditional top-down approaches used in manufacturing today's chips. Bottom-up manufacturing is of necessity a hierarchical process. First the individual devices and wires are manufactured. In this area there has been great success producing devices in quantity [12, 24, 31]. Then individual devices must be assembled into systems for device and circuit experiments [44, 43, 34]. To create successful molecular electronics systems we must be able to assemble, en masse, the individual components into larger subunits. These subunits would then be connected together into complete systems.

3.1 Techniques

There are many different techniques for assembling and aligning nanoscale components [42]. These include Langmuir-Blodgett films, flow-based alignment, nano-imprinting, electro-magnetic alignment, self-assembled monolayers, and catalyzed growth. The common feature of all the assembly techniques, with the possible exception of nano-imprinting, is they can only form regular structures. For example, masses of NTs and NWs have been made to self-assemble into regular arrays by fluidics [22, 35]. Irregular structures have been made using nano-imprinting, however, the process of creating the masters combined with contact printing may limit the smallest pitch to above 100nm [42].

NWs self-assemble into parallel arrays guided by fluid flow. An ethanol fluid carrying NWs in suspension is passed over the substrate guided through a channel in a mold. Average separation can be controlled by varying the flow rate and duration. The NWs adhere well to the silicon substrate, permitting multiple layers to be assembled. Arrays of NWs crossing at right angles have been made this way [22].

Large regular meshes are not sufficient to create logical circuits. One proposed method of creating aperiodic complex structure out of simple meshes is by programmably cutting wires [40]. Crossbar wires can be cut by applying a high voltage which over-oxidizes the crosspoint and breaks the NW, since it's small enough to be consumed by the chemical reaction. This is higher than the voltage used to open or close the crosspoint.

One potential alternative to the above approaches is to use complementary strands of DNA as a "smart glue." DNA directed synthesis has been used to connect both nanometer-sized components [29] and micron-sized components [28]. In both cases the key is that selective binding of components can be controlled by varying the length and coding of a DNA sequence attached to the components being linked together. DNA-based selective binding has also been used to algorithmically guide assembly and define differentiation points within an array at the nanometer scale [41].

3.2. Implications

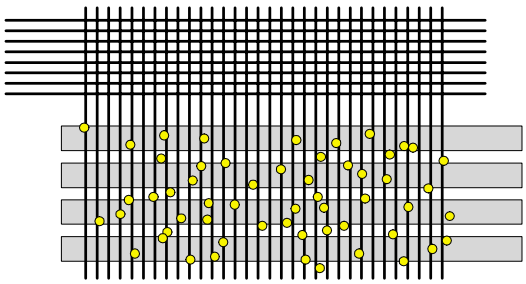
Bottom-up manufacturing, and in particular, the use of self-assembly as the dominant means of circuit construction imposes the most severe limitation on nanoscale architectures: precise device alignment will be difficult, or impossible, to achieve. Chemical self-assembly, as a stochastic process, will produce precise alignment of structures only rarely, and manipulation of single nanoscale structures to construct large-scale circuits is impractical at best. Furthermore, the defect rates will be high. Thus, bottom-up molecular-scale fabrication is universally expected to require defect tolerance due to the nature of chemical processes and alignment at that size.

Consequently, Molecular-scale architectures demand a different paradigm for system design than lithographic-scale architectures. In the lithographic era, we could rely on patterning to specify exactly where each active component and interconnect was placed. At the molecular scale, we can build regular, crystalline structures with small base feature size, but may not be able to deterministically make each lattice site or wire differ in a controlled manner at manufacturing time.

3.3. Structures

The available fabrication primitives imply three classes of architectures: fully deterministic, quasi-regular, and completely random. The fully deterministic approaches require precision and control that may one day be available with nano-imprinting or DNA-based assembly. Nano-imprint precision is likely to be expensive to build. The quasi-regular approaches assume the ability to assemble some regular structures (*e.g.*, two-dimensional meshes), using self-assembly techniques. Finally, the least structured approaches demand only the ability to randomly place molecules in a given area. All of these approaches will require some kind of defect tolerance. The latter two approaches also require that the desired functionality of the circuit be created post-fabrication. In other words, the fabricated circuits will be programmable logic, (*e.g.*, an FPGA).

Stochastic Assembly A good example of how random approaches can yield useful deterministic devices is a demultiplexor created from the random deposition of gold particles (see Figure 5) [39]. By controlling the chemistry, it is possible to grow a randomly distributed collection of gold particles between the input address wires and the molecular-scale nanowires or nanotubes. The set of connections to each nanowire acts as a code to select that nanowire. If we can arrange the code space and statistics of the random connections appropriately, we can arrange for most all of the core nanowires to each have a unique address code. For example, with 50% of the potential connections randomly connected, a code space with $4\log_2(N)$ address bits will allow one to uniquely address



A few lithographed input horizontal metal lines (bottom) connect to many vertical nanowires through randomly distributed gold particles. These nanowires form a programmable crossbar with many output horizontal nanowires (top). After testing, the crossbar is programmed to select the vertical nanowires which happen to form a binary demultiplexor. Placing parts with molecular-scale accuracy is not required.

Figure 5: Stochastically Assembled Demultiplexor

almost all of the N nanoscale wires. Once constructed, this demultiplexor can be used to allow a set of lithographic-scale wires to selectively address any of a large number of molecular-scale wires.

Another example of the class of random-based architecture is the nanocell architecture [23]. In the nanocell, gold particles coated with molecules with NDR behavior are randomly deposited within in a small square region (1 micron square). The particles create a random graph which can be probed to determine the functionality that was randomly created.

The common feature of these two structures is that the order necessary to create a useful logical device is discovered after the device is fabricated. Once the functionality is discovered, then the device is programmed to capitalize on what was created.

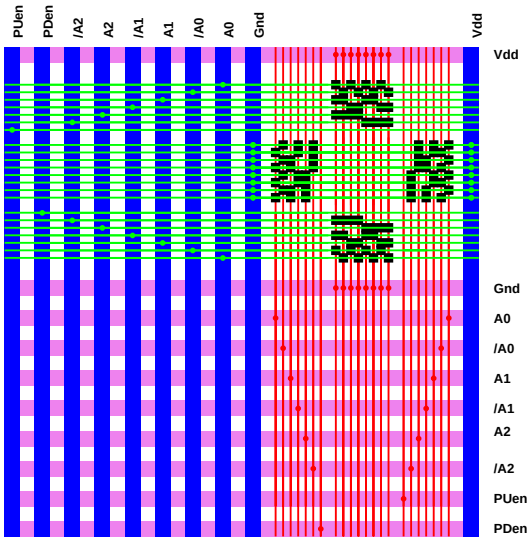
Quasi-Regular Assembly The quasi-regular class of architectures are typified by the hierarchical assembly of meshes. Each mesh consists of wires and programmable molecular devices. The meshes are then connected together into a larger mesh of configurable elements. Unlike the random-based architectures, the potential functionality of the device does not have to be discovered. However, these architectures do depend on post-fabrication programming in order to create logical circuits. Furthermore, since the fabrication primitives are unlikely to reliably yield perfect meshes these architectures require re-programmable components in order to provide defect-tolerance.

One of the difficulties in creating quasi-regular architectures is in addressing individual wires in the meshes from either the nanoscale or the micronscale. One method is to use the demultiplexor's described above. Once we have a way to address individual nanoscale wires, we can assemble these decoders with a crossed nanowire memory core, allowing us to uniquely address each crosspoint in the memory structure (See Figure 6). This same basic structure allows us to address and program crosspoint junctions which serve as wired-OR junctions in PLAs or crossbars.

4. Architectures

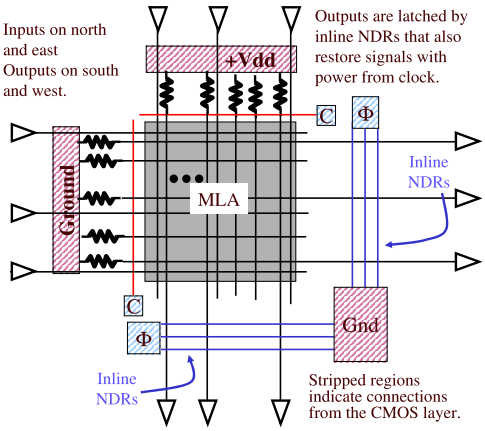
Architectures have now been proposed which suggest that these devices and assembly techniques are sufficient to build complete electronic systems which store and process information. We present two quasi-regular architectures.

Goldstein and Budiu describe an architecture, called a nanoFabric, in the quasi-regular class [19]. The nanoFabric is designed



Shown here is an 8×8 nanoscale wire array bracketed by the decoders used to program the array and connections to micronscale wires. As shown, the array is small compared to the micronscale wires. Note, however, that the number of micronscale wires scales logarithmically in array width; so for the larger nanoarray sizes we consider typical, the micronscale wiring becomes a thin periphery around a large nanoscale array core.

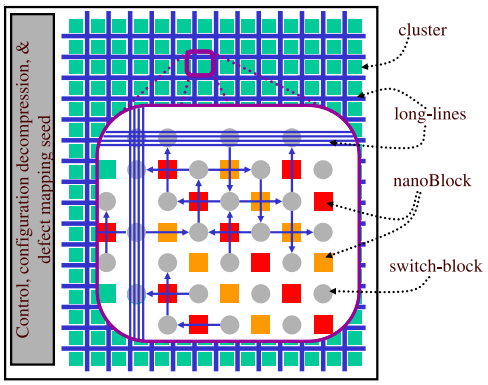
Figure 6: Array Bracketed with Decoders



The molecular logic array (MLA) is reconfigurable and supports the creation of diode-resistor based logic circuits. Signal restoration is performed using molecular latches which are orthogonal to the output lines of the MLA.

Figure 7: A nanoBlock

to overcome the limitations of chemical assembly. The basic unit of logic is a nanoBlock (See Figure 7). Each nanoBlock is based around a molecular logic array (MLA). At each intersection of the MLA is a reconfigurable switch (*e.g.*, a pseudo-rotaxene) in series with a diode. Diode-resistor logic is used to perform logical operations. To create a complete logic family, signals and their complements are brought into each circuit which produce both the desired functions and their complements. Logic values are restored using molecular latches [20]; which also provide a mechanism for latching values and isolating outputs from one nanoBlock from the inputs to another nanoBlock. The nanoBlocks are grouped together



The layout of a nanoFabric is similar to an island-style FPGA. Each cluster contains 128 nanoBlocks locally connected through switchblocks. The long-lines are connected to the nanoBlocks on the perimeter of the cluster. Underlying the nanoscale components is a lithographically created substrate providing power, ground, etc. It also contains circuits to aid in self-test, configuration management, etc.

Figure 8: nanoFabric Layout

into clusters and arranged so that the outputs of a nanoBlock intersect the inputs to two other nanoBlocks. The area where the wires for four nanoBlocks intersect (two provide outputs, two accept inputs) is called a switchblock (See Figure 8). The result is a two-dimensional mesh of nanoBlocks. Assuming a 100nm process for the lithographic-scale support structures, 128 nanoBlocks per cluster, and 30 long lines per channel, this design yields approximately 10^8nanoBlocks/cm. The design assumes chemical self-assembly of the nanoBlocks and deterministic placement of nanoBlocks on the supporting structure.

DeHon shows how to build a universal architecture based on Silicon Nanowire (SiNW) FETs (See Figure 9) [15]. SiNWs are organized into nano-arrays to perform wide-fanin logic functions. Typical nano-arrays are expected to be 100-1000 SiNWs tall and wide, balancing the needs to amortize out the cost of lithographic programming features and to contain defect rates. Each SiNW overlaps multiple nano-arrays to both perform logic and provide interconnect between the nanoarrays (See Figures 9 and 10). The SiNW-FETs provide signal restoration and isolation. Decoders associated with each nano-array allow it to be programmed from the lithographic-scale support wires. The resulting architecture can be viewed as an array of PLA blocks similar to CPLDs. At 10–20nm SiNW pitch, nano-arrays of 500 SiNW on a side would be 5–10μm on a side. Leaving space for lithographic scaffolding, a 1cm^2 IC could have 1/4-1 million such nano-arrays.

Both of these architectures exhibit many common features. The atomic unit of logic is based on 2-dimensional meshes. Connections between the meshes are through nanoscale wires; *i.e.*, there is no need to go from the nanoscale to the lithographic scale and back again within a circuit. Circuits are created by post-fabrication reconfiguration. Furthermore, the components of the circuits are formed from devices which perform the logical operations and separate devices which provide gain and I/O-isolation. The self-assembled molecular components are supported by lithographically created structures (*i.e.*, CMOS). Finally, the the wires in the rows (and the columns) are equivalent so defects can be avoided by swapping functionality between wires in a row (or a column).

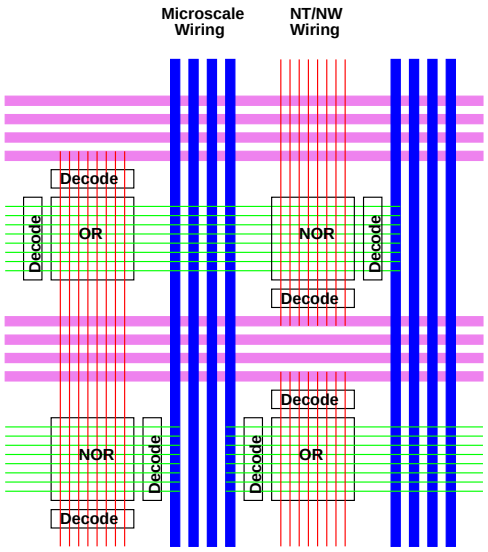
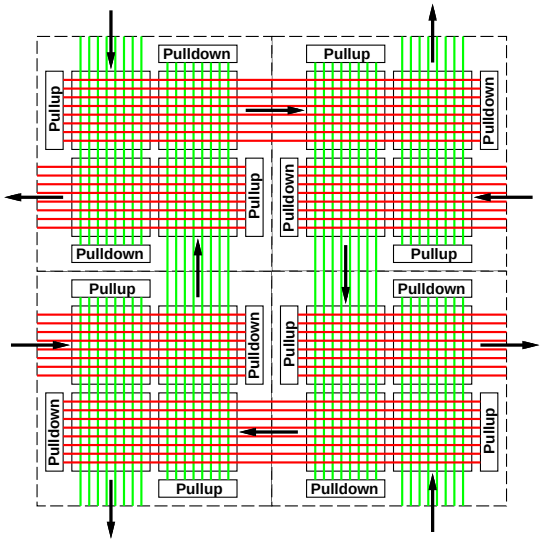


Figure 9: Assembly of Molecular-scale Array Logic into Computing System



We can build a logical NOR plane out of a 2×2 arrangement of crossed nano-arrays This arrangement allows inputs to enter from either side of the NOR-plane and outputs to depart in either orthogonal direction. Assembled into the macrotile shown, nano-array entry and exit freedom allows us to route signals in both dimensions, providing arbitrary Manhattan routing. This macrotile is abutted in both dimensions to build larger devices.

Figure 10: Macrotile for Routing and Logic

Molecular FPGA Area FPGA area is almost entirely determined by the area of its programmable crosspoints. Current technology uses about 20 crosspoint cells per ASIC gate-equivalent of logic, including interconnect. In lithographed CMOS, a minimal crosspoint cell is a 5T static memory cell and a large N-channel pass transistor. Since this takes much more area than the minimum wire pitch squared, crosspoint area determines FPGA logic density. Assuming 20 crosspoints per gate equivalent, in 45 nm lithography (ITRS 2010 node) FPGA density is about 4 million gates per cm^2. At 32 nm in 2013 it's 9 million gates, and at 22 nm in 2016, 18 million gates.

Molecular-scale programmable switches fit in the same area as the wire crossovers. This means logic and memory area are determined by the wire pitch. At the same conductor pitch, this makes programmable crosspoints in molecular electronics two orders of magnitude smaller than lithographic crosspoints. While molecular electronics may require programmable logic, it can still be far denser than in lithographed CMOS.

The molecular FPGA will have overhead for lithographic support, defect tolerance, and limits of quasi-regular device assembly. Conservatively assuming that this means we only use 20% of the device area for net useful crosspoints, a molecular FPGA with 50 nm wire pitch (about 10× nanowire thickness), would have 400 million gates per cm^2, which is 20× the gate density of the 22 nm CMOS FPGA of 2016. It is also roughly equivalent to the 360 million gates per cm^2 which the ITRS 2001 roadmap predicts for CMOS ASICs at the 32 nm node in 2013. Scaling down to 25 nm molecular wire pitches or higher utilizations may allow molecular FPGAs to surpass lithographic CMOS ASICs in density in the early 2010's.

5. Defect and Fault Tolerance

We expect non-trivial defect and fault rates in molecular electronics systems. Defects arise from bottom-up, chemical assembly of wires and devices. Fault rates increase at the molecular electronics scale because α-particles, long a concern of dynamic memory, can disrupt the logic of operating gates and may create new permanent defects in the logic. Further, since we are no longer dealing with large numbers of dopants, conduction paths, or electrons, bulk abstractions (*e.g.* current flow) and law of large numbers effects will break down. For example, statistical fluctuations in electron transfer times may cause variations in gate timing or behavior.

5.1. Defect Tolerance

The inevitable presence of defects in our circuits will drive both the selection of suitable organizations and the selection of appropriate, abstract models of device composition. In general, we expect to test fabricated devices to locate faulty components (without probing the individual components) and program the devices appropriately to avoid the known defects. HP's Teramac was a large-scale demonstration of the feasibility of this approach [13].

Memories, crossbars, and PLAs all have the convenient property that they do not depend on any particular line or crosspoint to function properly. Row and column sparing is heavily used in lithographic memories to avoid faults. The fact that programmable crosspoints in this technology are the size of a wire crossing makes crossbar solutions quite viable, allowing designs to locally switch around faulty wires or crosspoints. As long as arrays yield above a suitable fraction of wires in each dimension, they can be locally repaired in this manner. Arrays which yield below this fraction can be avoided completely, using alternate arrays at the device level.

The presence of defects will place demands at the system level. No two chips will be exactly alike. Local adaptation for faults within nano-arrays will be necessary. Nano-arrays themselves should be relocatable to account for defective array positions.

5.2. Fault Tolerance

For systems and architectures at this scale dynamic fault-tolerance will be necessary to address both transient faults and newly occurring defects. Systems will likely need a hierarchy of techniques to address these challenges drawn from the traditional fault tolerant literature and designed specifically for these systems. The literature on fault tolerance is beyond the scope of this tutorial.

6. CAD Challenges and Opportunities

Molecular electronics, due to its size and assembly methods presents new design automation problems which merit new classes of algorithms and optimizations. While many of these challenges are unique to molecular electronics, many of them are extreme instances of problems found in deep-submicron VLSI. Here we briefly present some of the challenges.

6.1. Testing

Traditionally, testing has been done to determine if the entire part is good or bad. With molecular electronics this will have to change. Although, traditional fault testing and signature analysis will be important, the defect densities and sheer number of devices on each part will make this a qualitatively different problem. Techniques such as those used in the Teramac form a basis for localizing faults without having to probe each individual device. New techniques, such as described in [30], suggest that defect rates of up to 10% can be localized in linear time, but these techniques demand new analysis methods and novel test circuits.

Couple the need for fault localization with low bandwidth to the outside world, both through the nano→lithographic interface and through the lithographic→board interface, and external testing alone will become completely prohibitive. Some testing will need to be integrated into the lithographic substrate (*e.g.*, see Figure 8). Furthermore, the test process will need to bootstrap the device so that it can ultimately test and configure itself.

6.2. Defect and Fault Tolerance

Using parts which have high fault rates and some defective components presents many new challenges to circuit designers and architects. Von Neumann's fault tolerance scheme [32] shows an existence proof for building robust circuits from unreliable gates. However, in the worst-case, its overhead is undesirably large. On the other hand, worst-case complexity circuits are seldom seen, thus, it is possible there are methods which can provide similar or tighter guarantees for typical circuits. Instead of fault observability, perhaps synthesis algorithms which guarantee fault non-observability are needed.

Another approach is to generate circuits with a fail-stop property. Such circuits, based on sparse and redundant encoding of state machines for example, could allow us to detect when faults have occurred and provide high likelihood that the execution will not continue. A more robust, but possibly harder approach, would be to compute in an alternative codespace, a codespace in which local operations could be performed with built-in error correction (*e.g.*, [37]). However, such codespaces do not obviously allow comparisons, thus we may, at the very least, decode data, perform an operation, and re-encode it. The challenge here, as above, is to implement the encoding and decoding operations such they are inexpensive in area or time. Lightweight encoding schemes and operations which can be carried out directly on encoded data could be very important here.

Tolerating defects and eliminating faults solely at the device or circuit level will be impractical. For example, just managing the number and kinds of defects will require a hierarchical approach. Thus, effective fault and defect containment may require a standard subblock size, above the individual gates or wires. The subblock could be the unit of local repair to hide defects and the unit of substitution when entire subblocks are not repairable. In the simplest cases, the logic is simply shuffled to skip defective wires and crosspoints in the subblock. In more sophisticated cases, a just-in-time

mapping may be performed for the subblock logic. The subblock needs to be small enough that interconnect delay does not dominate so that operations within the subblock occur in a single cycle. This has the effect that placement shuffling does not destroy local subblock timing. Such systems have been proposed for traditional systems [18, 38, 5] and nanoFabrics [19]. In [18] the system is composed of subblocks, each an FSM with datapath (FSMD), and memories. In [19], each subblock is a split-phase abstract machine (SAM), which includes local memory, an FSM, and a datapath. The entire circuit is divided into SAMs such that each operator in the SAM has a fixed, small latency. All operations with long or unpredictable latency are split between SAMs. SAMs are then a natural unit of fault containment, detection, and rollback.

6.3. Novel Devices and Circuits

The sheer number of different combinations of even the smallest organic molecules promises to provide a larger variety of active devices than has been traditionally available (*e.g.*, programmable switches as described in Section 2.5). Determining the best selection of devices, and how they will be incorporated into circuits is a new challenge. These devices will have novel functionality but may lack traditional device functions. For example, many of the devices do not provide gain. In order to use these successfully, they must amalgamated with devices that do provide gain. New methods for determining when and where to insert gain-providing, or isolating devices will be required for scalable circuit design.

6.4. Optimizing for Spatial Locality

With chips which are a 1000 or more nano-arrays wide running at high clock rates, it will likely take 1000s of cycles to cross between distant nano-arrays. This is the natural scaling of logic speed relative communications which is already forcing us to take multiple clock cycles to reach far points on silicon chips. Optimizing the placement of logical functions so that commonly communicating elements are close will be a first order concern for optimizing performance.

6.5. Scalable Place and Route

The sheer size of these systems makes placement and routing times a concern. Add to that the need to accommodate different defect locations for every component, and the problem becomes even more acute. The subblock approach mentioned in Section 6.2 may be one piece of the solution—hierarchically decomposing the problem into independent intra-subblock placement and routing and inter-subblock placement and routing. The small distances and regular structures inside the subblock may make intra-subblock placement and routing trivial (*e.g.* using wire interchangeability within a PLA or crossbar). Inter-subblock placement may be eased by exploiting delay independent design (Section 6.6) for inter-subblock communications; this allows tools to move individual subblocks (*e.g.*, SAMs) around without needing to re-place and re-route the intra-subblock connections. Going further, we might create architectures where the device itself can assist inter-subblock runtime placement and routing [16].

6.6. Communication Centric Design

As wires ever increasingly dominate design, communication centric design will become more important. This is, again, especially true if subblocks are going to be relocated to support defect and fault tolerance. Delay independent design will allow the computation to deal with varying delays that arise from placement and variable data arrival rates associated with filtering out corrupt data. Timing independence further opens the door to a wide range of communication schemes abstracted from the details of the end point computations and enables dynamic relocation of communicating subblocks to improve spatial locality.

Key components for communication centric design appear in Berkeley's SCORE model [5]. SCORE divides the computation into fixed-size compute pages which are connected together by persistent dataflow links, or *streams* to form a computational graph. SCORE allows compute graphs to have arbitrary size and to evolve during the computation. Data on the streams is tagged with presence, providing deterministic, timing-independent behavior for the graph.

6.7. Design Size

The design time for the billion-gate devices molecular electronics enables is also an important concern. If we assume 500 RTL gates per designer per day in 2000 [3], and apply the current 21% per year productivity growth rate identified by Sematech [27], then 2010 productivity is 3300 gates per designer-day. A billion-gate design will take 1200 designer-work-years, a completely impractical effort for all but a very few. While a factor of 10 gain in designer productivity will be required for end of the roadmap CMOS, a factor of 100 may be necessary to harness molecular hardware. Design productivity will be the gating resource for developing molecular hardware products.

Today we are applying standard platform designs, IP cores, and system-level design to increase productivity. This may get us a factor of ten eventually. Another factor of ten will be needed. Possible solutions include parallelism, spatial computations, and silicon compilation. Regular parallelism is a well known way to multiply area and design effort into performance. Billion gate designs can easily contain thousands of CPUs (even complex, 64-bit, million gate CPUs). Applications and algorithms with heavy, high-level parallelism will be increasingly favored. Software could be compiled directly into dedicated hardware. An adder for every '+, a mux for every 'if, etc., for entire major applications. Early research [2, 4] indicates MediaBench and SpecInt95 benchmarks would compile into several million bit-level operators.

7. Conclusions

Working molecular electronic devices exist today. Research progress is steady and strong, giving us cause to believe that molecular electronic systems may be practical in five to ten years. If lithography reaches fundamental physical or economic limits, molecular electronics may allow us to continue observing Moore's Law. Regardless, molecular bottom-up fabrication could give us a much better alternative, whose price would depend mainly on design and test cost, instead of billion-dollar factories.

Challenges to making this reality are plentiful at every level, some naturally in physics and chemistry, but many in ICCAD. These include fabricating and integrating devices, managing their power and timing, finding fault-tolerant and defect-tolerant circuits and architectures and the test algorithms needed to use them, developing latency-tolerant circuits and systems, doing defect-aware placement and routing, and designing, verifying and compiling billion-gate designs and the tools to handle them. Any one of these could block practical molecular electronics if unsolved.

Many of these are challenges that will be faced regardless of the underlying technology. Molecular electronics provides a pure and extreme example, and strengthens the case for solving them sooner rather than later.

Acknowledgments

Prof. Goldstein is funded by the DARPA Moletronics program under grants ONR N00014-01-0659 and MDA972-01-03-0005. Prof. DeHon is funded by the DARPA Moletronics program under grant ONR N00014-01-0651.

References

[1] Semiconductor Industry Association, "Int'l Technology Roadmap for Semiconductors," http://public.itrs.net/Files/2001ITRS/Home.htm, 2001.

[2] J. Babb, M. Rinard, C. A. Moriz, W. Lee, M. Frank, R. Barua, and S. Amarasinghe, "Parallelizing applications into silicon," in *Proceedings of the IEEE Symposium on Field-Programmable Custom Computing Machines (FCCM'1999)*, 1999, pp. 70–80.

[3] P. Bricaud and M. Keating, "IP Reuse Creation for System-on-a-Chip Design," in *Proceedings of the 1999 IEEE Custom Integrated Circuits Conference*, 1999.

[4] M.. Budiu, "Application-Specific Hardware: Computing Without CPUs," in *4th CMU Symposium on Computer Systems SOCS-4*, October 2001, also CMU Technical Report CMU-CS-01-164.

[5] E. Caspi, M. Chu, R. Huang, N. Weaver, J. Yeh, J. Wawrzynek, and A. DeHon, "Stream Computations Organized for Reconfigurable Execution (SCORE): Extended Abstract," in *Conference on Field Programmable Logic and Applications (FPL '2000)*. August 28–30 2000, LNCS, pp. 605–614, Springer-Verlag.

[6] J. Chen, M. Reed, A. Rawlett, and J. Tour, "Large On-Off Ratios and Negative Differential Resistance in a Molecular Electronic Device," *Science*, vol. 286, pp. 1550, 1999.

[7] J. Chen, W. Wang, M. A. Reed, M. Rawlett, D. W. Price, and J. M. Tour, "Room-Temperature Negative Differential Resistance in Nanoscale Molecular Junctions," *Appl. Phys. Lett.*, vol. 77, pp. 1224, 2000.

[8] C. Collier, G. Mattersteig, E. Wong, Y. Luo, K. Beverly, J. Sampaio, F. Raymo, J. Stoddart, and J. Heath, "A [2]Catenane-Based Solid State Reconfigurable Switch," *Science*, vol. 289, pp. 1172–1175, 2000.

[9] P. Collins, M. Arnold, and Ph. Avouris, "Engineering Carbon Nanotubes and Nanotube Circuits Using Electrical Breakdown," *Science*, vol. 292, pp. 706, 2001.

[10] Y. Cui, X. Duan, J. Hu, and C. Lieber, "Doping and electrical transport in silicon nanowires," *Journal of Physical Chemistry B*, vol. 104, no. 22, pp. 5213–5216, 2000.

[11] Y. Cui and C. Lieber, "Functional Nanoscale Electronic Devices Assembled Using Silicon Nanowire Building Blocks," *Science*, vol. 291, pp. 851, 2001.

[12] Y. Cui, C. Lieber, L. Lauhon, M. Gudiksen, and J. Wang, "Diameter-controlled synthesis of single crystal silicon nanowires," *Applied Physics Lett.*, vol. 78, no. 15, pp. 2214–2216, 2001.

[13] W.B. Culbertson, R. Amerson, R. Carter, P. Kuekes, and G. Snider, "Defect Tolerance on the TERAMAC Custom Computer," in *Proceedings of the IEEE Workshop on FPGAs for Custom Computing Machines*, Los Alamitos, California, April 1997, IEEE Computer Society, pp. 116–123, IEEE Computer Society Press.

[14] P. D. Franzon D. P. Nackashi, "Molectronics: A circuit design perspective," in *Proc. of the SPIE*, 2002, vol. 4236.

[15] A. DeHon, "Array-Based Architecture for Molecular Electronics," in *Proceedings of the First Workshop on Non-Silicon Computation (NSC-1)*, February 2002.

[16] A. DeHon, R. Huang, and J. Wawrzynek, "Hardware-Assisted Fast Routing," in *Proceedings of the IEEE Symposium on Field-Programmable Custom Computing Machines*, April 2002.

[17] J.C. Ellenbogen and J.C. Love, "Architectures for molecular electronic computers: 1. Logic structures and an adder designed from molecular electronic diodes," *Proc. IEEE*, vol. 88, no. 3, pp. 386–426, 2000.

[18] D. Gajski, N. Dutt, A. Wu, and S. Lin, *High-Level Synthesis: Introduction to Chip and System Design*, Kluwer Academic Publishers, 1992.

[19] S. Goldstein and M. Budiu, "NanoFabrics: Spatial Computing Using Molecular Electronics," in *Proc. 28th International Symposium on Computer Architecture*, June 2001.

[20] S. Goldstein and D. Rosewater, "Digital Logic Using Molecular Electronics," in *Int'l Solid State Circuits Conference*, Feb. 2002.

[21] Y. Huang, X. Duan, Y. Cui, L. Lauhon, K-H. Kim, and C. Lieber, "Logic Gates and Computation from Assembled Nanowire Building Blocks," *Science*, vol. 294, pp. 1313, 2001.

[22] Y. Huang, X. Duan, Q. Wei, , and C. Lieber, "Directed assembly of onedimensional nanostructures into functional networks," *Science*, vol. 291, pp. 630–633, January 2001.

[23] S. Husband, C. Husband, Price, Dirk, Franzon, Seminario, and Reed, "The Nanocell Approach to a Molecular Computer," Moletronics PI Meeting.

[24] S. Iijima, "Helical Microtubules of Graphitic Carbon," *Nature*, vol. 354, pp. 56, 1991.

[25] T.I. Kamins, R.S. Williams, Y. Chen, Y.-L. Chang, , and Y.A. Chang, "Chemical vapor deposition of Si nanowires nucleated by TiSi2 islands on Si," *Applied Physics Letters*, vol. 76, no. 562, 2000.

[26] R. Landauer, "Spatial Variation of Current and Fields Due to Localized Scatterers in Metallic Conduction," *IBM Journal of Research and Development*, vol. 1, no. 3, 1957.

[27] P. Magarshack, "Improving SoC Design Quality through a Reproducible Design Flow," *IEEE Design & Test of Computers*, vol. 19, no. 1, pp. 76–83, 2002.

[28] J. K. N. Mbindyo, B. R. Reiss, B. R. Martin, C. D. Keating, M. J. Natan, and T. E. Mallouk, "DNA-Directed Assembly of Gold Nanowires on Complementary Surfaces," *Advanced Materials*, vol. 13, pp. 249–254, 2001.

[29] C. Mirkin, "Programming the Assembly of Two- and Three-Dimensional Architectures with DNA and Nanoscale Inorganic Building Blocks," *Inorg. Chem.*, vol. 39, pp. 2258–72, 2000.

[30] M. Mishra and S. C. Goldstein, "Scalable Defect Tolerance for Molecular Electronics," in *First Workshop on Non-Silicon Computing*, Cambridge, MA, February 2002.

[31] A. Morales and C. Lieber, "A laser ablation method for the synthesis of crystalline semiconductor nanowires," *Science*, vol. 279, pp. 208–211, 1998.

[32] J. Von Neumann, "Probabilistic Logic and the Synthesis of Reliable Organisms from Unreliable Components," in *Automata Studies*, C. Shannon and J. McCarthy, Eds. Princeton University Press, 1956.

[33] D. J. Pena, B. Razavi, P. A. Smith, M. J. Natan, T. S. Mayer, T. E. Mallouk, and C. D. Keating, "Electrochemical Synthesis of Multi-Material Nanowires as Building Blocks for Functional Nanostructures," in *MRS Symposium Proceedings*, 2001, vol. 636, pp. D4.6.1–4.6.6.

[34] T. Rueckes, K. Kim, E. Joselevich, G. Tseng, C. Cheung, and C. Lieber, "Carbon nanotube based nonvolatile random access memory for molecular computing," *Science*, vol. 289, pp. 94–97, 2000.

[35] R. Service, "Assembling Nanocircuits From the Bottom Up," *Science*, vol. 293, pp. 782, 2001.

[36] H. Soh, C. Quate, A. Morpurgo, C. Marcus, J. Kong, and H. Dai, "Integrated nanotube circuits: Controlled growth and ohmic contacting of single-walled carbon nanotubes," *Applied Physics Letters*, vol. 75, no. 5, 1999.

[37] D. Spielman, "Highly Fault-Tolerant Parallel Computation," in *Proceedings of the 37th Annual IEEE Conference on Foundations of Computer Science 1996*, 1996.

[38] D. Sylvester and K. Keutzer, "Rethinking Deep-Submicron Circuit Design," *IEEE Computer*, vol. 32, no. 11, pp. 25–33, November 1999.

[39] S. Williams and P. Kuekes, "Demultiplexer for a Molecular Wire Crossbar Network," United States Patent Number: 6,256,767, July 3 2001.

[40] S. Williams, P. Kuekes, and J.R. Heath, "Molecular-Wire Crossbar Interconnect (MWCI) for Signal Routing and Communications," US Patent 6,314,019, November 6, 2001.

[41] E. Winfree, F. Liu, L.A. Wenzler, and N.C. Seeman, "Design and Self-Assembly of Two-Dimensional DNA Crystals," *Nature*, vol. 394, pp. 539–544, 1998.

[42] Y. Xia, J. Rogers, K. Paul, and G. Whitesides, "Unconventional Methods for Fabricating and Patterning Nanostructures," *Chem. Rev.*, vol. 99, pp. 823–1848, 1999.

[43] C. Zhou, M. R. Deshpande, M. A. Reed, and J. M. Jones II, L. anmd Tour, "Nanoscale Metal/Self-Assembled Monolayer/Metal Heterostructures," *Appl. Phys. Lett*, vol. 71, pp. 611, 1997.

[44] C. Zhou, C.J. Muller, M.R. Deshpande, J. W. Sleight, and M.A. Reed, "Microfabrication of a mechanically controllable break junction in silicon," *Applied Physics Letters*, vol. 67, pp. 1160, 1995.