Fault-Tolerant NanoBoxes for Designing Computers Using Molecular Nanotechnology

AJ KleinOsowski Richard A. Kiehl David J. Lilja
ajko@ece.umn.edu kiehl@ece.umn.edu lilja@ece.umn.edu
Department of Electrical and Computer Engineering
Minnesota Supercomputing Institute
University of Minnesota, Minneapolis, MN 55455

I. Introduction

Molecular computing, in which computer components are constructed from molecules, is a rapidly growing topic in the field of nanotechnology. To date, researchers have demonstrated molecular devices which can be controlled and which can store a state, equivalent to being able to control a transistor to store a single bit of information [2], [4], [7]. A variety of other types of molecular devices also may be possible [8]. Additionally, researchers have demonstrated nanoscale wires to connect the various devices together [3] and devices which perform logic [1], [5], [6]. Figure 1 shows a molecular approach for assembling nanoscale circuitry using DNA scaffolding [12]. The natural characteristics of these molecular devices, however, result in memory devices which unexpectedly and frequently change state.

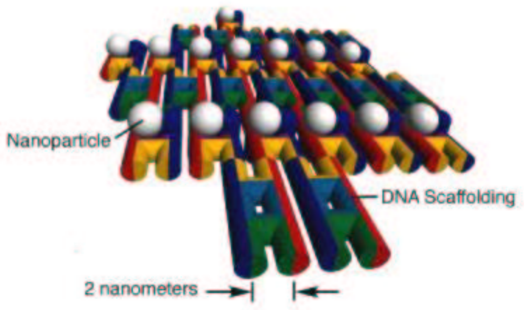

Fig. 1. One possible approach to fabricating molecular computers uses DNA scaffolding to assemble nanoscale components from molecules.

The small size of molecular nanodevices is a double-edged sword. On the one hand, the extremely small size of these devices means that we can pack over one trillion devices into a square centimeter of die space [14]. This device density is more than 100,000 times that of the transistor density of Intel's newest microprocessor chip, the Itanium, which packs 25 million transistors [13] into 300 square millimeters of die space [10]. The flip side of this unsurpassed device density is that these small devices, by nature of their size, are not able to drive much current. Furthermore, the manufacturing yield and the relability of molecular devices is expected to be much lower than conventional devices.

To build computers out of weak and unstable molecular devices, we need to invent clever ways to incorporate fault-tolerance into the system design without adversely affecting the overall performance, die space, and power consumption.

II. NanoBoxes

Prior work on building computers out of molecular devices [9], [11] used external testing hardware to periodically survey the system and identify faulty components. As suggested in Figure 2, the interconnections among the devices can be reconfigured to route around the faulty components. Unfortunately, it is not clear that these approaches will adequately scale to the large number of devices that will be available with this molecular nanotechnology.

We are beginning to investigate an approach for dealing with the high number of faulty devices by incorporating fault-tolerant features directly into "black box" finite state machines, as shown in Figure 3. These fault-tolerant components then can be used as the fundamental building blocks for computer systems. In these so-called *NanoBoxes*, faults are identified and corrected on the fly, thereby eliminating the need for external fault testing hardware.

Exactly how to build these fault-tolerant NanoBoxes is the focus of our preliminary work. Classical fault-tolerance techniques can be used as a springboard. How-

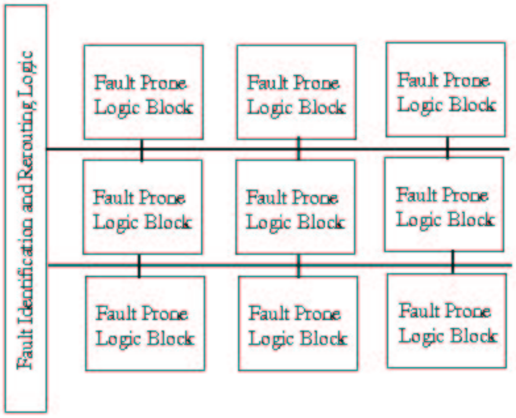

Fig. 2. Prior approaches for constructing fault tolerant architectures for nanodevices use external hardware to identify and route around faulty logic blocks.

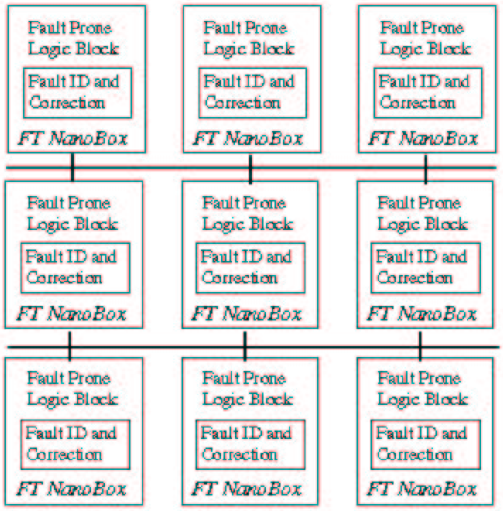

Fig. 3. The NanoBox architecture incorporates fault tolerant nanotechnology techniques into deterministic logic blocks. Signals are tested and verified before leaving the NanoBox.

ever, molecular nanodevices have several fundamental characteristics that are different from classical silicon-based devices. These differences will demand a new approach for incorporating fault-tolerance.

We envision a device in which all of the device replication and fault detecting communication occurs inside the NanoBox wall. Outside some as-yet-to-be-defined boundary, the inputs and outputs of the NanoBox are assumed to be deterministic and conventional. For instance, the external signals may change only on system-wide clock edges. With this partitioning, our NanoBoxes behave in ways that are conventional to both device physicists and computer architects. By veiling the details of fault-tolerance for unreliable molecular nanodevices, CAD systems can be built with varying levels of system component abstraction, thereby allowing computer architects to construct entire computer systems using classical finite state machine-based techniques.

While our ideas are very preliminary, we hope to encourage discussion and debate concerning the fundamental problem of fault-tolerance in molecular-level systems.

REFERENCES

[1] Yaakov Benenson, Tamar Paz-Elizur, Rivka Adar, Ehud Keinan, Zvi Livneh, and Ehud Shapiro. Programmable and autonomous computing machine made of biomolecules. *Nature*, 414:430–434, November 2001.

[2] Robert R. Birge. Protein-based optical computing and memories. *IEEE Computer*, 25(11):56–67, November 1992.

[3] Yong Chen, Douglas A. A. Ohlberg, Gilberto Medeiros-Ribeiro, Y. Austin Chang, and R. Stanley Williams. Self-assembled growth of epitaxial erbium disilicide nanowires on silicon (001). *Applied Physics Letters*, 76(26):4004–4006, June 2000.

[4] C.P. Collier, E.W. Wong, M. Belohradsky, F.M. Raymo, J.F. Stoddart, P.J. Kuekes, R.S. Williams, and J.R. Heath. Electrically configurable molecular-based logic gates. *Science*, 285:391–394, July 1999.

[5] Alberto Credi, Vincenzo Balzani, Steven J. Langford, and J. Fraser Stoddart. Logic operations at the molecular level: An XOR gate based on a molecular machine. *Journal of the American Chemical Society*, 119(11):2679–2681, 1997.

[6] A. Prasanna de Silva and Nathan D. McClenaghan. Proof-of-principle of molecular-scale arithmetic. *Journal of the American Chemical Society*, 122(16):3965–3966, April 2000.

[7] V. Derycke, R. Martel, J. Appenzeller, and Ph. Avouris. Carbon nanotube inter- and intramolecular logic gates. *Nano Letters*, 1(9), September 2001.

[8] David Goldhaber-Gordon, Michael S. Montemerlo, J. Christopher Love, Gregory J. Opiteck, and James C. Ellenbogen. Overview of nanoelectronic devices. *Proceedings of the IEEE*, 85(4):521–540, April 1997.

[9] Seth Copen Goldstein and Mihai Budiu. Nanofabrics: Spatial computing using molecular electronics. In *International Symposium on Computer Architecture (ISCA)*, July 2001.

[10] Linley Gwennap. *Intel's Merced and IA-64: Technology and Market Forecast*. MicroDesign Resources, 1999.

[11] James R. Heath, Philip J. Kuekes, Gregory S. Snider, and R. Stanley Williams. A defect-tolerant computer architecture: Opportunities for nanotechnology. *Science*, 280:1716–1721, June 1998.

[12] Richard A. Kiehl. Nanoparticle electronic architectures assembled by DNA. *Journal of Nanoparticle Research*, 2:331–332, 2000.

[13] Kevin Krewell. Intel raises the Itanium. *Microprocessor Report*, 15(6), June 2001.

[14] Vwani P. Roychowdhury, David B. Janes, and Supriyo Bandyopadhyay. Nanoelectronic architecture for boolean logic. *Proceedings of the IEEE*, 85(4):574–588, April 1997.