Electronic Nanotechnology Based on Non-Binary Principles

Jimson Mathew, Elena Dubrova and Hannu Tenhumen

Royal Institute of Technology, LECS/IMIT, 164 40 Kista, Sweden

ABSTRACT

Non-silicon based computing technologies open new possibilities for designing electronic circuits which employ more than two discrete levels of signal. Such circuits, called multiple-valued logic circuits, have a number of theoretical advantages over standard binary circuits. In this paper, we discuss possibilities for implementing multiple-valued functions using chemically assembled electronic nanotechnology. Simulation results for a three-valued case is also given.

Keywords: multiple-valued logic, nanotechnology, molecular diode, adder,CAEN.

1 INTRODUCTION

CMOS-based integrated circuit technology is built on the principle "a data bit is always zero or one". This is due in large part to the availability of simple, cheap and reliable CMOS transistors with just two stable states: on and off. However, novel computing technologies based on molecular electronics, quantum mechanics or biological processes open interesting possibilities for utilizing *multiple-valued logic* and alternative to binary number systems.

Multiple-valued logic is a generalization of classical Boolean logic. It provides a theoretical base for designing electronic circuits with more than two logic levels of signal. Multiple-valued logic circuits have a number of theoretical advantages over standard binary circuits. For example, on and off chip interconnect can be reduced if signals in the circuit assume four or more levels rather than only two [1]. In memory design, storing two instead of one bit of information per memory cell doubles the density of the memory in the same die size [2]. Applications using arithmetic circuits often benefit from using alternatives to binary number systems. For example, residue and redundant number systems can reduce or eliminate the ripple-through carries which are involved in normal binary addition or subtraction, resulting in high-speed arithmetic operations [3].

In spite of these potential advantages, practicality of multiple-valued logic design heavily depends on the availability of circuit realizations which must be competitive with present-day binary technologies. The attempts to built integrated circuits employing multi-stable state devices and signals with more than two discrete levels can be traced back to 1970, starting from the early works on ternary designs. Multiple-valued logic circuits have been implemented in bipolar technology, such as integrated injection logic (I^2L) and emitter-coupled logic (ECL), in complementary metal oxide semiconductor (CMOS) technology and in n-type MOS technology [5]. However, with exceptions of flash and DRAM memory applications [2], silicon-based multiple-valued designs are inferior to their binary counterparts [6].

The impressive advances of non-silicon based technologies in recent years brings new hopes for non-binary computing to become a reality. A number of fully functional computing devices employing more than two stable states have been already proposed, including resonant tunneling transistors (RTT), resonant tunneling diodes (RTD) [7], surface tunneling transistors (STT) [8]. Negative-differential-resistance characteristics which appear in these devices have clear multiple thresholds and therefore are very promising for non-binary computing [9].

The purpose of this paper is to stimulate the reader's interest beyond conventional binary case. We first give an introduction to multiple-valued logic theory. We then look at the possibilities for implementing multiple-valued logic circuits using molecular electronics. We show how chemically assembled electronic nanotechnology can be used for multiple-valued computing. An example of a three-valued adder implementation and simulation results are shown. Finally, we conclude with a discussion of some open problems.

2 CAEN

A promising alternative to CMOS-based technology is chemically assembled electronic nanotechnology (CAEN) [10]. CAEN exploits the quantum-mechanical effect of nanometer scale devices. The electronic circuits are constructed by self-alignment and self-assembly of these small devices rather than using lithography. A CAEN switch is a two-terminal device behaving like a diode. Compared to a CMOS transistor, such a switch uses much less power, since only a few electrons are used for switching. Moreover, a two-terminal device

can be implemented in CAEN with inexpensive chemical assembly. On the other hand, three-terminal electronic nanotechnology devices require precise alignment to allocate three wires at the device, which makes them unsuitable for producing real circuits [10]. The logic functions are implemented in CAEN using two-terminal configurable switches and diodes only. A consequence is that no inverters can be build, and therefore all logic signals should be available in both non-complemented and complemented form to perform computations.

Due to the constraints associated with the direct assembly of nanometer-scale components, CAEN technology is unsuitable for fabricating non-regular architectures. In [10], an regular architecture called *nanoFabric*, allowing to exploit the advantages of molecular electronics was introduced. The nanoFabric is a two-dimensional homogeneous array of interconnected *nanoBlocks*, which can be configured to implement any logic function. Reconfigurability of nanoFabric is essential for handling fabrication defects. Since CAEN-based devices have a much higher defect density than CMOS-devices, their yield is very low. Thus, post-fabrication self-diagnosis should first be applied to locate the defects. Then, the fabric is configured to implement the desired functionality by routing around the defected spots.

A nanoBlock is a square-shaped molecular array which can be programmed to implement a 3-input 3-output Boolean function and its complement. The inputs are placed on north and west sides of the array and the outputs are taken from the south and east sides of the array. The functions are implemented by programming the configurable switches which lie at each intersection of two orthogonal wires. NanoBlocks can also be used as switches for routing. In addition to local routing between the nanoBlocks, there are long-lines that run between the clusters of nanoBlocks to provide communication over longer distances and promote scalability.

3 MULTIPLE-VALUED FUNCTIONS IN CAEN

Next, we show how nanoFabric can be used to implement multiple-valued functions. A number of possibilities for such a generalization exist, depending on which properties of nanoFabric we would like to preserve. We have chosen to keep the following features of nanoFabric:
(1) the architecture is a two-dimensional homogeneous array;
(2) it can be configured to implement any logic function;
(3) only two-terminal devices are used, i.e. diode-resistor logic;

The third point implies that, as in binary case, no inverters are available. First, we discuss what is "inverter" in multiple-valued case.

In binary case, inverter implements the complement operation defined by $x' = 1 - x$, where x is a Boolean variable $x \in \{0,1\}$. It can easily be generalized to the complement of a multiple-valued variable $x \in M, M = \{0,1,2,\ldots,m-1\}$ as $x' = (m-1) - x$. The AND operation can be extended to a MIN, whose output is the smaller of the two values of its variables. The set of operations {AND, NOT} is known to be functionally complete for Boolean functions $\{0,1\}^n \rightarrow \{0,1\}$. However, the extended set {MIN, NOT} is *not* functionally complete for multiple-valued functions $M^n \rightarrow M$. In fact, it is not possible to extend the binary NOT to any single unary operation which would result in a functionally complete system in combination with MIN [17]. To achieve completeness of an m-valued system, a variable $x \in M$ should have $m-1$ different "complements" rather than just one. One such unary operations is *literal*, defined as follows.

$$x^S = \begin{cases} 1 & \text{if } x \in S \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where $S \subseteq M$.

In binary case, a consequence of the absence of inverters is that all logic signals should be available in both complemented and non-complemented form [10]. In multiple-valued case, the absence of literal gates implies that the signals for all literals have to be available. For example, if $m = 3$, then a three-valued variable x should be available as x^0, x^1 and x^2. Similarly, the output function f has to be represented as f^0, f^1 and f^2. This way of computing is convenient because a literal is a characteristic function of type $M \rightarrow \{0,1\}$. Therefore, the operations on literals are Boolean operations of type $\{0,1\}^n \rightarrow \{0,1\}$. For example, a sum of two literals $x^i + x^j$ or a product of two literals $x^i \cdot x^j$ are binary Boolean operations of type $\{0,1\}^2 \rightarrow \{0,1\}$. Hence, we can define the sum "+" as OR and the product "$\cdot$" as AND and represent multiple-valued functions as sum-of-products of literals. Next, we show that these three operations are sufficient to define an arbitrary multiple-valued input binary-valued output function of type $M^n \rightarrow \{0,1\}$. The proof can be found in [17, chapter 4].

First, we observe that the constants 0 and 1 can be defined as $0 = \prod_{j \in M} x_i^j$ and $1 = \sum_{j \in M} x_i^j$, for any $i \in \{1,2,\ldots,n\}$. Next, we extend the Shannon decomposition theorem from binary to multiple-valued case as follows.

Theorem 1 *Every multiple-valued input binary-valued output functions $M^n \rightarrow \{0,1\}$ can be decomposed with respect to a variable x_i, $i \in M$, as*

$$f(x_1,\ldots,x_n) = \sum_{j=0}^{m-1} x_i^j \cdot f|_{x_i=j}$$

where $f|_{x_i=j} = f(x_1, \ldots, x_{i-1}, j, x_{i+1}, \ldots, x_n)$ *are cofactors of* f *with respect to* x_i, $\sum$ *stands for OR and "·" stands for AND.*

By subsequently applying Theorem 1, we can decompose an n-variable function with respect to all its variables and derive the following canonical sum-of-products expression. Let "+" = OR, "·" = AND and $L = \{x^0, x^1, \ldots, x^{m-1}\}$ be the set of literals specified by equation (1).

Theorem 2 *Every multiple-valued input binary-valued output function* $M^n \to \{0,1\}$ *has a canonical expression in terms of* $\{+, \cdot, L\}$ *of type*

$$f(x_1, \ldots, x_n) = \sum_{i=0}^{m^n-1} c_i \; x_1^{i_1} \; x_2^{i_2} \ldots \; x_n^{i_n}$$

where $c_i \in \{0,1\}$ *are binary constants, and* $(i_1 i_2 \ldots i_n)$ *is the* m*-ary expansion of* i *with* i_1 *being the least significant digit.*

The set M of m elements together with the set A of functions over M and two distinct elements $\mathbf{0}, \mathbf{1}$ in M generate a finite algebra $\langle M; A; \mathbf{0}, \mathbf{1}\rangle$. For example, the set $B = \{0,1\}$ with the functions AND, OR and NOT generate the Boolean algebra $\langle B; +, \cdot, '; 0, 1\rangle$. The constants $\mathbf{0}$ and $\mathbf{1}$ are two particular elements of M which have special properties. They are called the *zero* and the *unit* of the algebra, respectively. The following result follows directly from Theorem 2.

Theorem 3 *The algebra* $\langle M; +, \cdot, L; \mathbf{0}, \mathbf{1}\rangle$, *where "+" is OR, "·" is AND,* $L = \{x^0, x^1, \ldots, x^{m-1}\}$ *is the set of literals,* $\mathbf{0} = 0$ *and* $\mathbf{1} = 1$, *is functionally complete for multiple-valued input binary-valued output functions* $M^n \to \{0,1\}$.

It follows from Theorem 3, that we can implement any multiple-valued function using the diode-resistor logic similar to the one used for Boolean functions in [10]. The difference is that, instead of using the complemented and non-complemented values of the signals, we are going to use the literals $x^0, x^1, \ldots, x^{m-1}$ of the input variables and the literals $f^0, f^1, \ldots, f^{m-1}$ of the output functions. Since, for all $i \in M$, f^i is of type $M^n \to \{0,1\}$, it can always be expressed in terms of AND, OR and literals of the variables $x_1, \ldots, x_n$. Furthermore, since all signals for the literals of the variables are available, any f^i can be implemented by the molecular logic array, similar to the binary molecular logic array proposed in [10].

The implementation of a three-valued adder in the molecular logic array is shown in Figure 1. Figure 2 shows the simulation waveforms. We used Molspice simulation tool from CMU with HSPICE as back-end. Molspice is a Java based front-end for simulating molecular circuits. It has some simplified simulation built in, but uses SPICE as a back-end for more accurate simulation. The first two sets of waveforms show the three-valued inputs X and Y. The last two sets show the sum and carry outputs. The three possible values of X input are shown as {(100), (010), (001)} which represents 0, 1 and 2. From the first set of waveform in the Figure 2 it can be seen that only one of the inputs is high at a time. Values for Y input are shown in second set of waveforms. To make the diagrams more readable, we show only a sample (3 out of 9) of all possible combinations of values for X and Y inputs. The respective sum and carry outputs of the above said inputs are shown in the last two sets. The third value of the carry is not shown because it is always zero. In order to verify the results, let us consider the inputs (001)(2) and (001)(2) for X and Y. The corresponding sum and carry are (010) (1) and (010) (1), which are the correct values for sum and carry outputs in a three valued adder. It is important to note that the sum and carry output voltages are not fully restored. For voltage restoration molecular latches described in [15] can be used. Molecular latches are implemented using resonant tunneling diodes, which provides the most important benefits of the lithographically fabricated transistor [15].

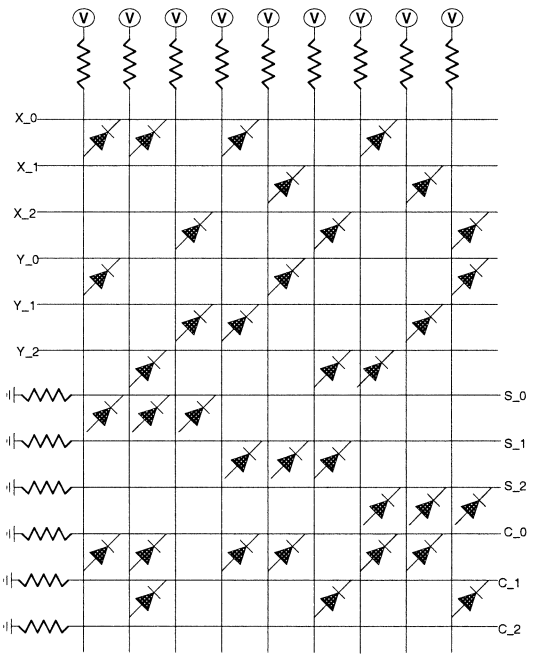

Figure 1: A three-valued adder implemention using molecular logic array.

4 CONCLUSION

We gave a brief introduction to the theory of multiple-valued logic related to the design of electronic circuits: multiple-valued functions, functionally complete sets and multiple-valued algebraic systems. We discussed possi-

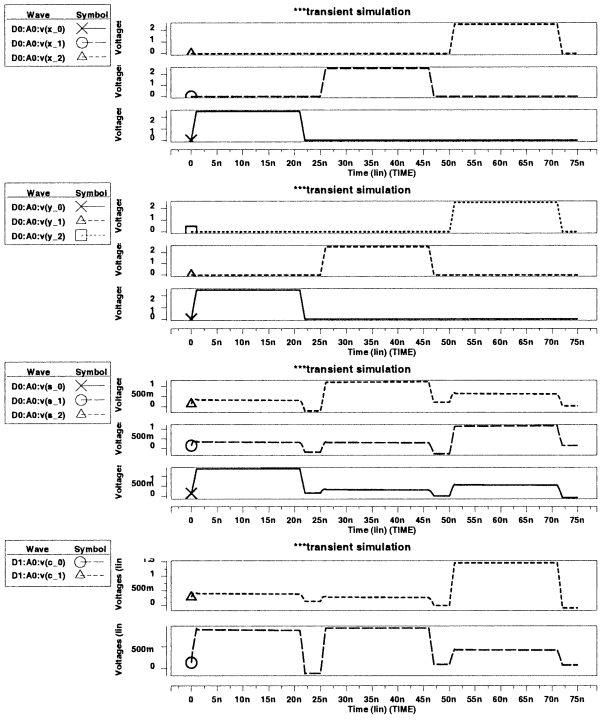

Figure 2: Three-valued adder using molecular logic array: simulation waveforms.

bilities for implementing multiple-valued functions with chemically assembled electronic nanotechnology. Furthermore, simulation results for three-valued adder is described. For full voltage restoration at the output molecular latches can be used.

A challenge to practical utilization of multiple-valued logic is creation of an effective computer-aided design package. Although some of the concepts and algorithms necessary for such a package have been already developed [18], a significant amount of research remains to be done to make it competitive with standard CAD packages.

Acknowledgment

The authors would like to thank Prof. S.C. Goldstein, School of Computer Science, Carnegie Mellon University, Pittsburgh, USA for providing us with Molspice simulator. This work was supported in part by the Research Grant No 6415 from Socware Research and Education.

REFERENCES

[1] I. Ben Dhaou, E. Dubrova and H. Tenhunen, Power efficient inter-module communication for digit-serial architectures in deep-sub-micron technology, *Proc. 30th Int. Symp. Multiple-Valued Logic*, (2000), 61-67.

[2] B. Ricco, G. Torelli, M. Lanzoni, A. Manstretta, H. E. Maes, D. Monatanari and A. Modelli, Non-volatile multilevel memories for digital applications, *Proc. IEEE*, **86**, 12, (1998), 2399-2421.

[3] T. Hanyu, M. Kameyama, A 200 MHz pipelined multiplier using 1.5 V-supply multiple-valued MOS current-mode circuits with dual-rail source-coupled logic, *IEEE Journal of Solid-State Circuits*, **30**, 11, (1995), 1239-1245.

[4] K. Shimabukuro, C. Zukeran, Reconfigurable current-mode multiple-valued residue arithmetic circuits, *Proc. 28th Int. Symp. Multiple-Valued Logic*, (1998), 282-287.

[5] E. Dubrova, Multiple-Valued Logic in VLSI Design, *Multiple-Valued Logic, An International Journal*, (2002).

[6] D. Etiemble, M. Israel, Comparison of binary and multi-valued ICs according to VLSI criteria, *Computer* **21**, 4 (1988), 28-42.

[7] T. Itoh et al., 10G Hz operation of multiple-valued quantizers using resonant tunneling diodes, *IEICE Trans. Inf. & Syst.*, **E82-D**, (1999), 949-954.

[8] T. Uemura, T. Baba, Demonstration of a novel multiple-valued T-gate using multiple-junctions surface tunnel transistor and its applications to three-valued data flip-flop, *Proc. 30th Int. Symp. Multiple-Valued Logic*, (2000), 305 -310.

[9] W.-C. Liu et al., Multiple negative differential resistance phenomena of metal-insulator-semiconductor-insulator-metal (MISIM)-like structure with step-composed $In_xGa_{1-x}As$ quantum wells, *IEEE Trans. on Electron Devices*, **45**, 2, (1998), 373-379.

[10] S. C. Goldstein, M. Budiu, NonFabrics: Spatial computing using molecular electronics, *Proc. of the 28th Annual Int. Symp. on Computer Architecture*, (2001)

[11] S. Wei, K. Shimizu, Residue arithmetic multiplier based on the radix-4 signed-digit multiple-valued arithmetic circuits, *12th Int. Conf. on VLSI Design*, (1999), 212-217.

[12] O. Ishizuka, D. Handoko, VLSI design of a quaternary multiplier with direct generation of partial products, *Proc. 27th Int. Symp. Multiple-Valued Logic*, (1997), 169-174.

[13] A. F. Gonzalez, P. Mazumder, Multiple-valued signed digit adder using negative differential resistance devices, *IEEE Trans. on Computers*, **47**, 9, (1998), 947 - 959.

[14] J. C. Muzio and T. C. Wesselkamper, *Multiple-valued switching theory*, Bristol, Hilger, 1986.

[15] S. C. Goldstein, D. Rosewater, Digital Logic Using Molecular Electronics, *ISSCC '02, San Francisco, CA,*, (2002)

[16] E. L. Post Introduction to a general theory of elementary propositions, *Amer. J. Math.*, **43**, (1921) 163-185.

[17] S. Hassoun and T. Sasao, eds., *Logic Synthesis and Verification*, Kluwer Academic Publishers, 2002.

[18] M. Gao, J.-H. Jiang, Y. Jiang, Y. Li, S. Sinha and R. Brayton, MVSIS, in *Proc. Int. Workshop on Logic Synthesis*, (2001), 138-144.