Proceedings of the IASTED International Conference
BIOMEDICAL ENGINEERING
February 16-18, 2005, Innsbruck, Austria

MICRO-/NANOFLUIDIC COMPUTING AND ITS APPLICATIONS TO BIOMEDICINE

Toshinori Munakata, Computer and Information Science Dept.
Cleveland State University, Cleveland, OH 44115, U.S.A.
t.munakata@csuohio.edu

David W.M. Marr, Chemical Engineering Dept.
Colorado School of Mines, Golden, CO 80401, U.S.A.
dmarr@mines.edu

ABSTRACT
We have recently begun investigation of a non-silicon based new computing paradigm called "micro-/nanofluidic computing" where basic computing elements such as logic gates may be incorporated into very small scale devices. The two key aspects of micro-/nanofluidics are the extremely small amount of materials processed and the relatively fast processing time. Because of these characteristics, micro-/nanofluidics is envisioned to have significant application potential in many areas including biomedicine and computation.

With these advantages, micro-/nanofluidic computing is not intended to replace traditional silicon-chip computers because its speed is orders of magnitude slower. Rather, it is envisioned to enable nanofluidic technology by directly incorporating computing functions. Here, the current state and our recent work in micro-/nanofluidic computing are introduced. In addition, advantages and potential applications specific to biomedicine are discussed.

KEY WORDS
Microtechnology, BioMEMS, Bio-nanotechnology, Bioinformatics, and Micro-/nanofluidic computing.

1. Introduction

For the past 40 years computer hardware has been dominated by silicon-based technology. Recently new computing paradigms other than silicon chips have been developed. While extensive practical use of these as computing devices is yet to be seen, these ideas have stimulated the scientific community due to their fundamental nature, novelty, and potential for new forms of information processing and applications. They include quantum, DNA, optical, chaos, molecular/atomic scale, and micro-/nanofluidic computing:

Quantum computing was introduced by Richard P. Feynman at Caltech in 1982. He pointed out that certain quantum mechanical phenomena are difficult to simulate by traditional Turing computers. To overcome these difficulties, he suggested the use of computers based on quantum mechanics. In a quantum computer, each quantum bit or qubit can be 0, 1 or in a superposition of states – 0 and 1 at the same time. This means that a qubit can encode more information than a classical bit. Such quantum computing has been applied to computationally difficult problems such as search, cryptography, and number theory [1, 2].

DNA computing or more generally *biological computing* was first developed by Leonard M. Adleman at USC in 1994 where information is encoded on DNA which is then used to perform bio-molecular processes to achieve targeted computing. As in the case of quantum computing, it has been applied to computationally difficult problems. Its single element speed is slow, on the order of 10^{-3} sec, but the use of massively parallel elements may provide a significantly faster overall effective speed [3, 4].

Optical computing has been around since the 1970’s and employs lasers to realize logic gates, leading to potentially high speeds. Research in this field slowed down during the 80’s and early 90’s but has since revived during the late 90’s with the development of new laser materials [5].

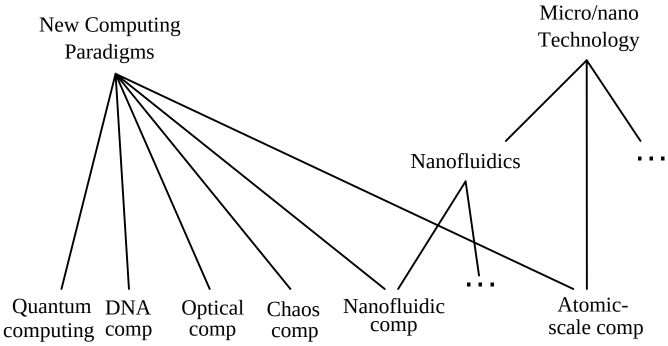
Chaos computing, a term coined by one of the authors of this paper, Toshinori Munakata, was demonstrated in 1998 with the construction of a NOR gate using a chaotic system by Sudeshna Sinha and William Ditto. Later all of these authors demonstrated that all the basic ingredients of computing - other logic gates, adders and memory – can be constructed with a chaotic or nonlinear system. To date, chaos computing has been primarily theoretical but has significant potential for very fast information processing [6].

Molecular and atomic scale computing has seen significant development recently where the goal is to realize computing functions directly at the nanoscale using carbon nanotubes for example. In this, a computer could be constructed by assembling these nanoscale elements using a bottom-up approach in contrast to the top-down design in traditional silicon-based computers [7, 8].

Micro-/nanofluidic computing [9]. For the past decade, small fluidic devices have been developed [10, 11, 12, 13, 14] that control liquids on a scale from approximately 100 microns (1 micron = 10^{-6} m) down to 50 nanometers. Micro-/nanofluidics, hereafter called "nanofluidics" for simplicity, has many potential applications in areas such as biomedicine because of its size and short processing times.

Nanofluidic computing requires association of basic computing elements in a nanofluidic device for the construction of logic gates for logic operations, adders for arithmetic operations and memory to store information. As a sub-area of nanofluidics and nanotechnology, nanofluidic computing can be viewed as a special case of the new computing paradigms as depicted in the tree shown in Fig. 1.

Figure 1. Trees representing hierarchical structures of new computing paradigms and micro/nanotechnology.



Three major objectives of this paper are:

1. To introduce the field of nanofluidic computing and our recent work in this area.
2. To describe the advantages of nanofluidic computing such as the increased efficiency and accuracy of information transfer within an integrated nanofluidic system without external electronic devices.
3. To discuss its potential application to biomedicine.

2. Nanofluidic Computing: Basics

Because of its potential there has been significant recent interest in nanofluidic computing. This has led to fluid-based memory [15, 16] and attempts at logical/arithmetic operation based on electrochemical reactions [17]. Complementing these investigations is a new approach as illustrated in Fig. 2 and shown in Fig. 3 [9] where we have implemented basic logic gates such as AND, OR, and NAND on a microfluidic device. This design is based on the relative resistance of fluid flows to create these logic gates.

Figure 2. Design for logic gates. The two channels at the bottom right are for inputs, where each input can be either 0 (represented by white) or 1 (represented by light color). The top channels are, from the left to the right, outputs of the AND, OR, XOR, NAND and NOR logic gates. The dark dyed color represents 1, and light color 0.

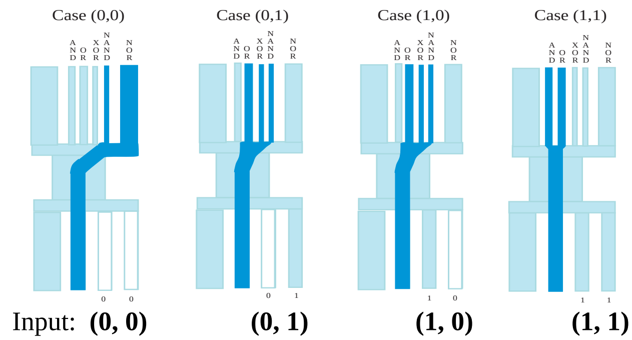


Figure 3. Experimental realization of the illustration of Figure 2. Only the output channels of Fig. 2 are shown. The diameter of each channel is approximately 100 microns. Channel structures could be scaled down to 50 nm using related lithographic techniques.

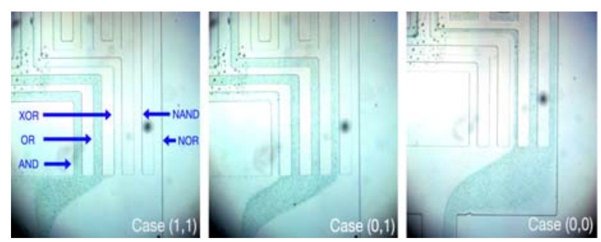


Table 1. Truth table for two-input logic gates

I_1, I_2	AND	OR	NAND	NOR	XOR
0, 0	0	0	1	1	0
0, 1 or 1, 0	0	1	1	0	1
1, 1	1	1	0	0	0

Table 1 is a truth table for some basic logic gates with two inputs I_1, I_2. Although not shown here, we have also implemented the single input gate NOT, where NOT (0) = 1 and NOT (1) = 0 [9]. Combined, these logic gates constitute the necessary ingredients of a computer, including logic/arithmetic units and memory.

One unique feature of the implementation demonstrated in Figs. 2 and 3 is that a number of logic gates are realized simultaneously in the same unit. This not only allows flexible selection of desired gates for a specific application, but it can also serve to construct a compound operation by combining some of these gates.

For example, one can implement a half adder by using the XOR gate to represent a significant bit and the AND gate as a carry bit.

One drawback of the specific approach depicted by Figs. 2 and 3 is that the output cannot be used to directly actuate subsequent fluid-based logic operators, limiting the direct scaling of the technique for nanofluidic computers. The approach, however, can be used with any technique or sensor that modifies the subsequent flow resistance within a channel, such as the integration of environmentally sensitive gels that could turn the input flows on or off within nanofluidic networks. In addition to the output-to-input transfer, an amplifier would significantly aid future design by maintaining information flow throughout a network.

3. Nanofluidics

Advantages

1. Very small quantities are sufficient for analysis and syntheses. For example, with fluidic channel height/width 1000 times smaller than a macroscopic device (e.g., 1 micron vs. 1 mm), the volume will be 10^6 times smaller. Hence, the technique is especially useful for expensive, hard-to-obtain or hazardous materials.
2. Processing time for analysis and syntheses is typically much shorter than equivalent macroscale processes because heat and mass transport occurs over extremely small length scales.
3. Parallel and complex analyses can be simultaneously performed by placing hundreds of nanochannels with unique environments on a small planar surface. This is analogous to computer integrated circuits where many circuits are integrated on a small silicon chip. In fact, the same basic technique for silicon chip fabrication is typically used to create nanofluidic channels. This high degree of parallelism can also contribute to overall faster processing times.
4. Handling techniques are relatively easy to learn.

Specific applications in biomedicine

o Blood tests.
Requires very tiny samples for automated multi-purpose, parallel processing analyses, and the result may be obtained within a fraction of a second. Current test tube scale analyses will become obsolete.

o Drug development and testing.
Each analysis requires very small amounts.
It is therefore cost effective.
Hazardous samples can be dealt with less risk.
Hundreds of tasks can be performed in parallel.
Faster development and cost effective.

o Homeland security
Analysis of trace of chemical/biological substance can be made rapidly.

4. Nanofluidic Computing: Characteristics

As stated earlier, although nanofluidic computing is one of the new computing paradigms, it will not replace traditional silicon-chip computers. The speed of nanofluidic computing will be much slower, 1 vs. 10^{10} operations per sec for a typical silicon-based computer. Thus, it is not desirable to use nanofluidic computing solely as a computing device. Rather, nanofluidic computing is envisioned to enhance nanofluidic technology through direct incorporation of computing functions in a device with another primary function.

Our goal therefore is to increase the efficiency and accuracy of information transfer and processing within integrated nanofluidic systems without the use of external electronic devices. As an alternative approach, a nanofluidic system may be connected to a traditional computer for computation (a fluidic-silicon hybrid environment). In this, data is measured from the system and fed into a computer, computation is performed, and then output sent back to the system. Which approach, stand-alone fluidic or hybrid, is better depends on many factors. Table 2 provides rule-of-thumb comparisons for the two approaches:

Table 2. Nanofluidic stand-alone vs. nanofluidic/silicon hybrid computing

	Stand alone	Hybrid
Information transfer	better	
Cost	better	
Space requirements	better	
Speed	depends (see *)	

*Speed characteristics depend on the amount and nature of computation; e.g., if extensive and/or complex, hybrid would be better. Examples of complex computations are statistical analysis, numerical differentiation/integration, algorithms involving searching, sorting, network graphs, etc.

5. Nanofluidic Computing: Applications

Potential application of nanofluidics for computing are broad and include:

Clinical and pathological testing.
Studies in genomics and proteomics.
Drug development and testing.

A typical scenario involving nanofluidic computing

During a nanofluidic process, one has an intermediate result, e.g., the existence of a certain chemical or biological component, or a physical quantity exceeds a threshold value as $x > x^*$. These can be expressed as a boolean $A = 1$ where non-existence or $x \le x^*$ is expressed as $A = 0$. Given other boolean measures such as B, C, etc. one takes a certain action, D. For example, $D = 1$ may represent the dispensing of a certain chemical and $D = 0$ not to dispense.

As an illustration, we consider a scenario. If (A = 1 AND B = 1 AND C = 1) OR (A = 1 AND at least one of B OR C = 1), then NOT D. The first step would be to employ boolean algebra to express the problem and simplify the expression if possible. In our case, the "if" part can be expressed as:

$$ABC + A(B + C)$$

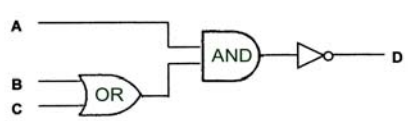
where multiplication represents logic AND and addition logic OR. This expression can be simplified as:

$$= A(BC + B + C) = A(B + C).$$

Such a problem could be implemented employing nanofluidic logic gates as shown in Fig. 4.

This type of unit could be combined with other units, forming a network where each unit may perform compound logical operations as illustrated above, as well as arithmetic operations and temporarily storing information.

Figure 4. Logic gate diagram for boolean expression D = NOT (A(B + C)).



Potential application examples

o Pathology

Cancer is a leading cause of death especially in developed countries. Many types of cancers are not only difficult to treat but also patients are diagnosed in advanced stages. Earlier detection and treatment of cancers will significantly increase the survival rates and well-being of patients. Since no two patients are exactly the same, individualized diagnosis and treatment are essential.

In a hypothetical scenario nanofluidic computing may be integrated with other techniques such as genomic chips that can identify the existence of certain genes. Other techniques may include the integration of nanoscale laboratories such as blood and cell tests. Perhaps 15 specific genes, G_1, ..., G_{15} and 10 lab test results, L_1, ..., L_{10}, are crucial for the diagnosis/treatment of a particular cancer, where $G_1 = 1$ if G_1 exists, and $G_1 = 0$ otherwise. Combinations of these parameter values are expressed as boolean expressions which are used for subsequent boolean operations and diagnosis/treatment. For example, if $G_1 = 0$ AND $G_2 = 0$ AND $G_3 = 0$ AND ($L_1 = 0$ OR $L_2 = 0$), then no cancer and no further investigation is necessary. If $G_1 = 1$ AND ($G_2 = 1$ OR $G_3 = 1$) AND ($L_1 = 1$ OR $L_2 = 1$), then proceed further to check G_5, G_8, L_3 and L_7. If ..., dispense drug D_1 from channel C_1 and D_2 from C_2, and send the result to C_3. These are extended versions of Fig. 4 that could be implemented in a nanofluidic computing environment.

o Development of a new drug

Most pharmaceutical research and development is based on finding slightly improved variants of patented drugs rather than developing purely novel compounds. Finding drug variants involves chemists synthesizing and testing hundreds of compounds, requiring an enormous cost in materials, time and manpower. The average cost today of developing a single new drug is approximately $300 million. To avoid such costs, a pharmaceutical company wishes to employ nanofluidic computing with a high degree of parallelism for much faster and cost effective drug development.

The entire process consists of hundreds of smaller processes connected as a directed network. A portion of the network can be described as follows. Decision D to perform a specific action is determined by the outcome of its preceding processes A, B, and C as follows: If $A = 1$ and at least one of B or $C = 1$, then do not proceed, i.e., $D = 0$, otherwise $D = 1$. This decision process is also represented by Fig. 4.

6. Conclusion

Two immediate technical issues to be resolved for nanofluidic computing are the connection of logic gates and the design of an amplifier. With solutions to these and related issues, extensive future development of nanofluidic computing for highly integrated nanofluidic chips is expected.

References:

[1] R. Feynman, Quantum mechanical computers, *Foundations of Physics, 16*, 6, 507-531,1986. Also in *Optics News, 11*, 1985.

[2] M.A. Nielsen and I.L. Chuang, *Quantum computation and quantum information* (Cambridge, UK: Cambridge Univ. Press, 2000).

[3] L.M. Adleman, Molecular computation of solutions to combinatorial problems, *Science, 266*, 1021-1024, Nov. 11, 1994.

[4] S.A. Kurtz, S.R. Mahaney, J.S. Royer, and J. Simon, Biological computing, in *Complexity theory retrospective II*, L.A. Hemaspaandra, and A.L. Selman, Eds., 179-195 (New York, Springer-Verlag, 1997).

[5] H. Abdeldayem, D. O. Frazier, M. S. Paley, and W. K. Witherow, Recent advances in photonic devices for optical computing, http://science.nasa.gov/headlines/images/nanosecond/thepaper.pdf.

[6] T. Munakata, S. Sinha, and W.L. Ditto, Chaos computing: Implementation of fundamental logical gates by chaotic elements, *IEEE Trans. Circuits and Systems I*, *49*, 2002, 1629-1633.

[7] H.C. Manoharan, C.P. Lutz, and D.W. Eigler, Quantum mirages formed by coherent projection of electronic structure, *Nature*, *403*, 512-515, 2000.

[8] A. Bachtold, P. Hadley, T. Nakanishi and C. Dekker, Logic circuits with carbon nanotube transistors, *Science*, *294*, 1317-1320, 2001.

[9] T. Vestad, D.W.M. Marr and T. Munakata, Flow resistance for microfluidic logic operations, *Applied Physics Letters*, *84*(25), June 21, 2004, 5074-5075.

[10] G.M. Whitesides and A.D. Stroock, Flexible methods for microfluidics, *Physics Today*, *54*(6), 2001, 43-48.

[11] A. Terray, J. Oakey, and D.W.M. Marr, Fabrication of linear colloidal structures for microfluidic applications, *Applied Physics Letters*, *81*, 2002, 1555-1557.

[12] A. Terray, J. Oakey, and D.W.M. Marr, Microfluidic control using colloidal devices, *Science*, *296*, 2002, 1841-1844.

[13] Marc A. Unger, Hou-Pu Chou, Todd Thorsen, Axel Scherer, and Stephen R. Quake, Monolithic microfabricated valves and pumps by multilayer soft lithography, *Science*, *288*, 2000, 113-116.

[14] David J. Beebe, Jeffrey S. Moore, Joseph M. Bauer, Qing Yu, Robin H. Liu, Chelladurai Devadoss and Byung-Ho Jo, Functional hydrogel structures for autonomous flow control inside microfluidic channels, *Nature*, *404*, 2000, 588-590.

[15] T. Thorsen, S. J. Maerkl, and S.R. Quake, Microfluidic large-scale integration, *Science, 298*, 2002, 580-584.

[16] A. Groisman, M. Enzelberger, and S.R. Quake, Microfluidic memory and control devices, *Science 300*, 2003, 955-958.

[17] W. Zhan and R.M. Crooks, Microelectrochemical logic circuits, *J. Am. Chem. Soc*., *125*, 2003, 9934-9935.