ELectronics EXpress

IEICE Electronics Express, Vol.3, No.13, 292–298

Design methodologies for compact logic circuits based on collision-based computing

Kazuhito Yamada[1], **Ikuko N. Motoike**[2], **Tetsuya Asai**[1a)], **and Yoshihito Amemiya**[1]

[1] *Graduate School of Information Science & Technology, Hokkaido University, Kita 14, Nishi 9, Kita-ku, Sapporo 060–0814, Japan*

[2] *Future University - Hakodate, 116–2 Kamedanakano-cho Hakodate 041–8655, Japan*

a) *asai@sapiens-ei.eng.hokudai.ac.jp*

Abstract: A method of designing compact multiple-input combinational logic circuits is proposed. We show that i) fundamental logic gates can be constructed by a small number of collision-based fusion gates, ii) multiple-input logic gates are constructed in a systematic manner, iii) the number of transistors in specific logic gates constructed by the proposed method is significantly smaller than that of conventional logic gates.

Keywords: collision-based computing, fusion gate, integrated circuits, logic circuits

Classification: Integrated circuits

References

[1] I. S. Reed, "A Class of Multiple-Error-Correcting Codes and Their Decoding Scheme," *IRE Trans. Inform. Theory*, vol. PGIT-4, pp. 38–49, 1954.

[2] D. E. Muller, "Application of Boolean Algebra to Switching Circuit Design and to Error Detection," *IRE Trans. Electr. Comput.*, vol. EC-3, pp. 6–12, 1954.

[3] M. Song and K. Asada, "Design of low power digital VLSI circuits based on a novel pass-transistor logic," *IEICE Trans. Electron.*, vol. E81-C, no. 11, pp. 1740–1749, 1998.

[4] M. Alioto and G. Palumbo, Model and Design of Bipolar and MOS Current-Mode Logic: CML, ECL and SCL Digital Circuits, Springer, 2005.

[5] H. Soeleman, K. Roy, and B. C. Paul, "Robust subthreshold logic for ultra-low power operation," *IEEE Trans. VLSI Syst.*, vol. 9, no. 1, pp. 90–99, 2001.

[6] R. S. Shelar and S. S. Sapatnekar, "BDD decomposition for the synthesis of high performance PTL circuits," *Workshop Notes of IEEE IWLS 2001*, pp. 298–303.

[7] N. Asahi, M. Akazawa, and Y. Amemiya, "Single-electron logic systems based on the binary decision diagram," *IEICE Trans. Electron.*, vol. E81-C, no. 1, pp. 49–56, 1998.

© IEICE 2006

DOI: 10.1587/elex.3.292
Received May 26, 2006
Accepted June 15, 2006
Published July 10, 2006

[8] A. Adamatzky, Editor, Collision-Based Computing, Springer-Verlag, 2002.
[9] A. Adamatzky A., B. De Lacy Costello, and T. Asai, *Reaction-Diffusion Computers.*, Elsevier, UK, 2005.

1 Introduction

Present digital VLSI systems consist of a number of combinational and sequential logic circuits as well as related peripheral circuits. A well-known basic logic circuit is a two-input NAND circuit that consists of four metal-oxide semiconductor field-effect transistors (MOS FETs) where three transistors are on the current path between the power supply and the ground. Many complex logic circuits can be constructed by not only populations of a large number of NAND circuits but also special logic circuits with a small number of transistors (there are more than three transistors on the current path) compared with NAND-based circuits.

A straight-forward way to construct low-power digital VLSIs is to decrease the power-supply voltage because the power consumption of digital circuits is proportional to the square of the supply voltage. In complex logic circuits, where many transistors are on the current paths, the supply voltage cannot be decreased due to stacking effects of transistors' threshold voltages, even though the threshold voltage is decreasing as LSI fabrication technology advances year by year. On the other hand, if two-input basic gates that have the minimum number of transistors (three or less) on the current path are used to decrease the supply voltage, a large number of the gates will be required for constructing complex logic circuits.

The Reed-Muller expansion [1, 2], which expands logical functions into combinations of AND and XOR logic, enables us to design 'specific' arithmetic functions with a small number of gates, but it is not suitable for arbitrary arithmetic computation. Pass-transistor logic (PTL) circuits use a small number of transistors for basic logic functions but additional level-restoring circuits are required for every unit [3]. Moreover, the acceptance of PTL circuits into mainstream digital design critically depends on the availability of tools for logic, physical synthesis, and optimization. Current-mode logic circuits also use a small number of transistors for basic logic, but their power consumption is very high due to the continuous current flow in turn-on states [4]. Subthreshold logic circuits where all the transistors operate under their threshold voltage are expected to exhibit ultra-low power consumption, but the operation speed is extremely slow [5]. Binary decision diagram logic circuits are suitable for next-generation semiconductor devices such as single-electron transistors [6, 7], but not for present digital VLSIs because of the use of PTL circuits.

To address the problems above concerning low-power and high-speed operation in digital VLSIs, we describe a method of designing logic circuits with collision-based fusion gates, which is inspired by collision-based reaction-

© IEICE 2006
DOI: 10.1587/elex.3.292
Received May 26, 2006
Accepted June 15, 2006
Published July 10, 2006

diffusion computing (RDC) [8, 9]. In the following sections, we introduce a new interpretation of collision-based RDC, especially concerning directions and speeds of propagating information quanta. We also show basic logical functions constructed by collision-based fusion gates, and discuss the number of transistors in classical and fusion-gate logic circuits.

2 Collision-based computing for digital VLSIs

Adamatzky proposed how to realize arithmetical scheme using wave fragments traveling in a reaction-diffusion medium where excitable chemical waves disappear when they collide each other [8, 9]. His cellular-automaton model mimicked localized excitable waves (wave fragments) traveling along columns and rows of the lattice and along diagonals. The wave fragments represented values of logical variables where logical operations were implemented when wave fragments collided and were annihilated or reflected as a result of the collision. One can achieve basic logical gates in the cellular-automaton model, and build an arithmetic circuit using the gates [8].

The cellular-automaton model for basic logic gates has been implemented on digital VLSIs [9]. Each cell consisted of several tens of transistors and was regularly arranged on a 2D chip surface. To implement a one-bit adder, for example, by collision-based cellular automata, at least several tens of cells are required to allocate sufficient space for the collision of wave fragments [8]. This implies several hundreds of transistors are required for constructing just a one-bit adder. Direct implementation of the cellular automaton model is therefore a waste of chip space, as long as the single cell space is decreased to the same degree of chemical compounds in spatially-continuous reaction-diffusion processors.

What happens if wave fragments travel in 'limited directions instantaneously'? Our possible answers to this question are depicted in Fig. 1. Figure 1 (a) shows our 2-in 2-out (C22) and 2-in 1-out (C21) units representing two perpendicular 'limited' directions of wave fragments, i.e., North-South and West-East fragments. The number of MOS transistors in each unit is written inside the black circle in the figure. The input fragments are represented by values A and B where A (or B) = '1' represents the existence of a wave fragment traveling North-South (or West-East), and A (or B) = '0' represents the absence of wave fragments. When A = B = '1' wave fragments collide at the center position (black circle) and then disappear. Thus, East and South outputs are '0' because of the disappearance. If A = B = '0', the outputs will be '0' as well because of the absence of the fragments. When A = '1' and B = '0', a wave fragment can travel to the South because it does not collide with a fragment traveling West-East. The East and South outputs are thus '0' and '1', respectively, whereas they are '1' and '0', respectively, when A = '0' and B = '1'. Consequently, logical functions of this simple 'operator' are represented by $\overline{A}B$ and $A\overline{B}$, as shown in Fig. 1 (a) left. We call this operator a collision-based 'fusion gate', where two inputs correspond to perpendicular wave fragments, and one (or two) output represents the re-

© IEICE 2006
DOI: 10.1587/elex.3.292
Received May 26, 2006
Accepted June 15, 2006
Published July 10, 2006

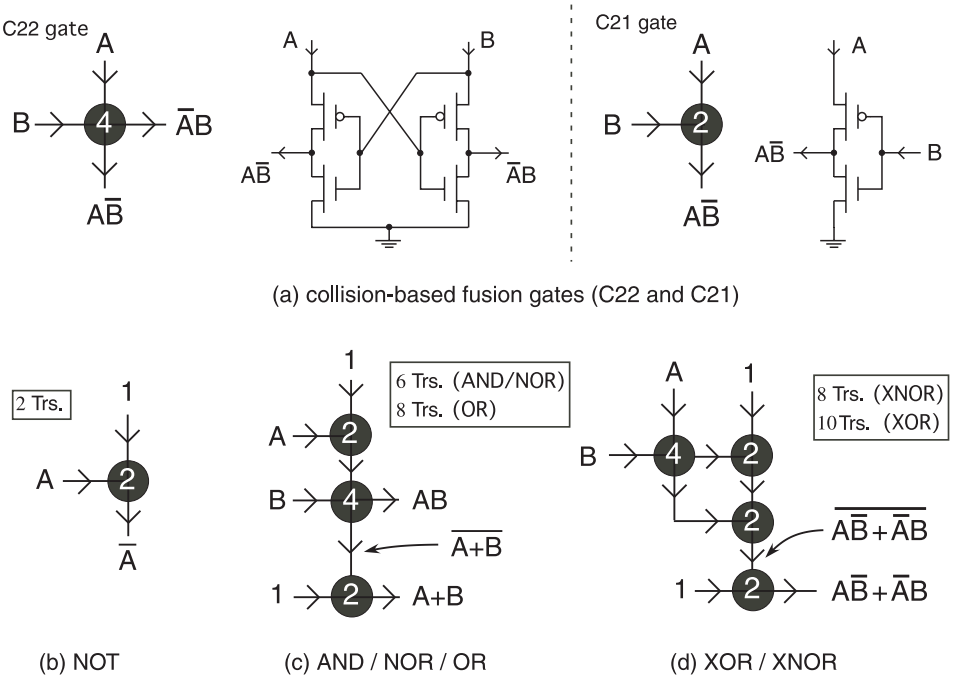

Fig. 1. Construction of collision-based fusion gates. (a) Definition of 2-in 2-out (C22) and 2-in 1-out (C21) gates and their corresponding circuits; (b) NOT; (c) AND, NOR, and OR; (d) XOR and XNOR functions.

sults of the collision (transparent or disappear) along the perpendicular axes. Figures 1 (b) to (d) represent basic logic circuits constructed by combining several fusion gates. The simplest example is shown in Fig. 1 (b) where the NOT function is implemented by a C21 gate (2 transistors). The North input is always '1', whereas the West is the input (A) of the NOT function. The output appears on South node ($\overline{A}$). Figure 1 (c) represents a combinational circuit of three fusion gates (two C21 gates and one C22 gate) that produces AND, NOR, and OR functions. Exclusive logic functions are produced by three (for XNOR) or four (for XOR) fusion gates as shown in Fig. 1 (d). The number of transistors for each function is depicted in the figure (inside the white boxes).

A collision-based fusion gate receives two logical inputs (A and B) and produces one (C21) or two (C22) logical outputs; i.e., $A\overline{B}$ for C21, $\overline{A}B$ and $A\overline{B}$ for C22. Unit circuits for C22 and C21 gates receive logical (voltage) inputs (A and B) and produce these logic functions. The minimum circuit structure is based on PTL circuits where a single-transistor AND logic is fully utilized. For instance, in Fig. 1 (a) right, a pMOS pass transistor is responsible for the $\overline{A}B$ function, and an additional nMOS transistor is used for discharging operations. When the pMOS transistor receives voltages A and B at its gate and drain, respectively, the source voltage approaches $\overline{A}B$ at equilibrium. If a pMOS transistor is turned off, an nMOS transistor connected between the pMOS transistor and the ground discharges the output node, which significantly increases the upper bound of the operation frequency. When A = B =

© IEICE 2006
DOI: 10.1587/elex.3.292
Received May 26, 2006
Accepted June 15, 2006
Published July 10, 2006

'0', the output voltage is not completely zero because of the threshold voltage of the MOS transistors, however, this small voltage shift is restored to logical '0' at the next input stage. Therefore additional level-restoring circuits are unnecessary for this circuit.

Figure 2 shows constructions of multiple-input logic functions with fusion gates. In classical circuits, two-input AND and OR gates consist of six transistors. As introduced in section 1, to decrease the power supply voltage for low-power operation, a small number of transistors (three or less) should be on each unit's current path. Since each unit circuit has six transistors, n-input AND and OR gates consist of $6(n-1)$ transistors ($n \geq 2$). On the other hand, in fusion gate logic [(a) and (b)], a n-input AND gate consisted of $4(n-1)$ transistors, whereas $2(n+1)$ transistors were used in an n-input OR gate. Therefore, in case of AND logic, the number of transistors in fusion gate circuits is smaller than that of classical circuits. The difference will be significantly expanded as n increases. Figure 2 (c) shows fusion gate implementation of majority logic circuits with multiple inputs. Again, in classical circuits, the number of transistors on each unit's current path is fixed to three. For n-bit inputs (n must be an odd number larger than 3), the number of transistors in the classical circuit was $30 + 36(n-3)^2$, while in the fusion gate circuit, it was $2(n^2 - n + 1)$, which indicates that the collision-based circuit has a great advantage in the number of transistors. Half- and full adders constructed by fusion gate logic are illustrated in Figs. 2 (d) and (e). The number of transistors in a classical half adder was 22, while it was

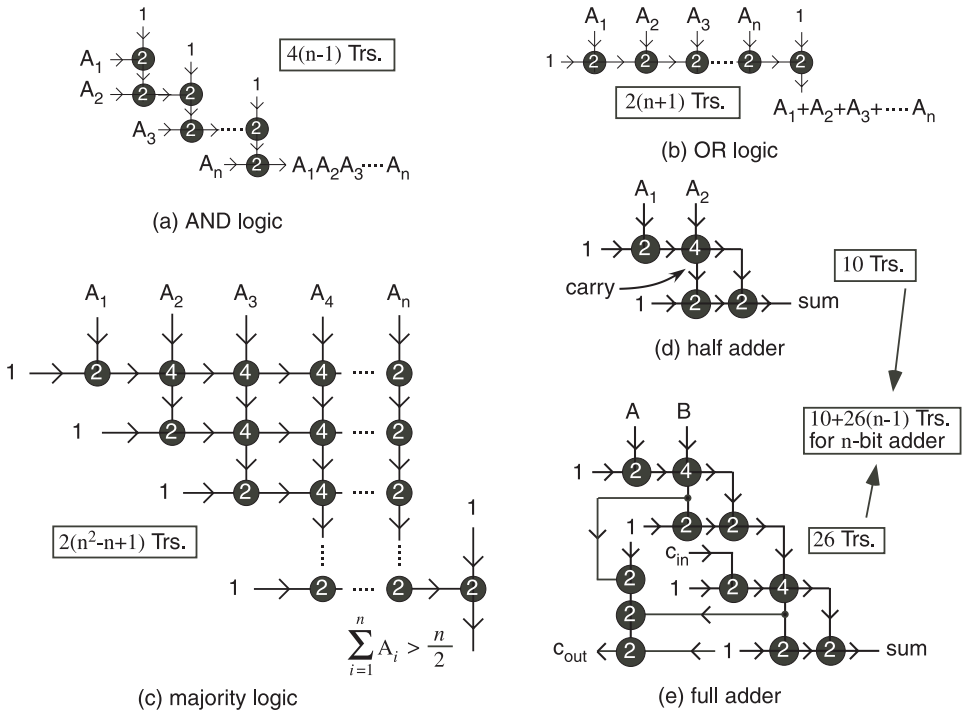

Fig. 2. Fusion gate architectures of multiple-input functions; (a) AND, (b) OR, (c) majority logic gates. Half and full adders are shown in (d) and (e), respectively.

© IEICE 2006
DOI: 10.1587/elex.3.292
Received May 26, 2006
Accepted June 15, 2006
Published July 10, 2006

10 in a fusion gate half adder [Fig. 2 (d)]. For n-bit full adders ($n \geq 1$), the number of transistors in a classical circuit was $50n - 28$, while it was $26(n-1)+10$ in a fusion gate circuit [Fig. 2 (e)]. Again, the fusion gate circuit has a significantly smaller number of transistors, and the difference will be increased as n increases.

Figure 3 summarizes the comparison of the number of transistors between classical and fusion gate logic. The number of transistors in fusion gate logic was always smaller than that of transistors in classical logic circuits, especially in majority logic gates.

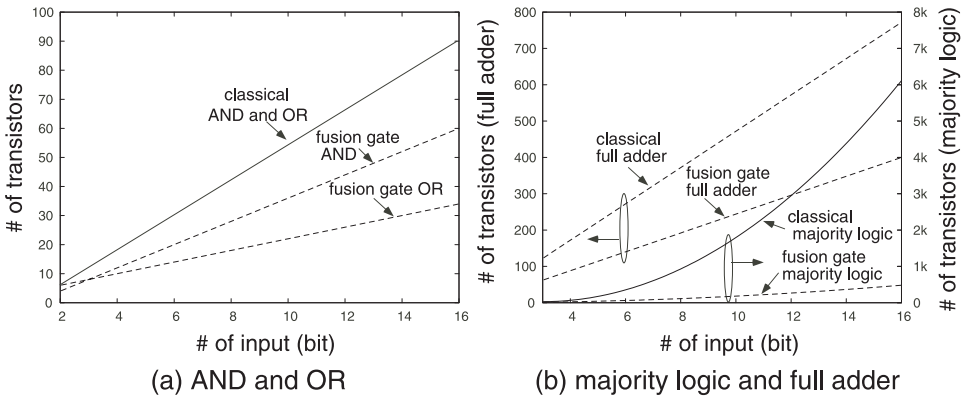

Fig. 3. Total number of transistors in classical and collision-based (CB) multiple-input logic gates.

3 Summary

We described a method of designing logic circuits inspired by collision-based reaction-diffusion computing. First, we introduced a new interpretation of collision-based computing, especially concerning a limited direction of wave fragments and infinite transition speed. This simplified constructions of the computing media significantly. Second, we showed that basic logical functions were able to be represented in terms of our unit operator, i.e., a 'fusion gate', that calculated $\overline{A}B$ and $A\overline{B}$ for inputs A and B. Third, basic MOS circuits for the fusion gate that consisted of two or four transistors were introduced. Then we demonstrated that in case of multiple-input logic functions, the number of transistors in fusion gate circuits is smaller than that of classical circuits, and the difference will significantly be expanded as n increases. Since the combination of the fusion gates produces multiple functions, e.g., an AND circuit can compute NOR simultaneously, we should build optimization theories for generating multiple-input arbitrary functions.

Acknowledgments

The authors wish to thank Professor Andrew Adamatzky of the University of the West of England for valuable discussions and suggestions during the research, and Professor Masayuki Ikebe of Hokkaido University for sugges-

© IEICE 2006
DOI: 10.1587/elex.3.292
Received May 26, 2006
Accepted June 15, 2006
Published July 10, 2006

tions concerning various CMOS circuits. This study was partly supported by Industrial Technology Research Grant Program in '04 from New Energy and Industrial Technology Development Organization (NEDO) of Japan, and a Grant-in-Aid for Young Scientists [(B)17760269] from the Ministry of Education, Culture Sports, Science and Technology (MEXT) of Japan.

© IEICE 2006
DOI: 10.1587/elex.3.292
Received May 26, 2006
Accepted June 15, 2006
Published July 10, 2006