IEICE TRANS. ELECTRON., VOL.E89–C, NO.11 NOVEMBER 2006

INVITED PAPER *Special Section on Novel Device Architectures and System Integration Technologies*

Single-Electron Logic Systems Based on a Graphical Representation of Digital Functions

Yoshihito AMEMIYA[†a)], ***Member***

SUMMARY This paper outlines the method of constructing single-electron logic circuits based on the binary decision diagram (BDD), a graphical representation of digital functions. The circuit consists of many unit devices, BDD devices, cascaded to build the tree of a BDD graph. Each BDD device corresponds to a node of the BDD graph and operates as a two-way switch for the transport of a single electron. Any combinatorial logic can be implemented using BDD circuits. Several subsystems for a single-electron processor have been constructed using semiconductor nano-process technology.

key words: *single electron circuit, logic gate, binary decision diagram, quantum dot, nanostructure*

1. Introduction

One of the goals in single-electron technology is the development of computing systems that can perform digital processing by making use of single-electron phenomena. There are two basic ways that we can use single-electron phenomena for digital processing. One is to construct transistor-like devices based on single-electron phenomena and imitate existing silicon LSIs, using the devices as analogs of MOS FETs. The problem with this approach is that integrated circuits composed of new transistor-like devices would have a hard time competing with the original silicon LSI, which is a product of well-established, mature technologies. The more promising approach is to reconsider the procedure for implementing digital processing and take up a way different from that of existing LSIs. From this approach, we will be able to construct novel computing systems that make good use of the properties of single-electron phenomena.

Various methods for digital processing are known for now. The procedure used in existing LSI systems is as follows.

- Devise an algorithm for processing a given information;
- Execute each step of the algorithm, under von Neumann-type computing architecture;
- In the execution, express the algorithm in the form of a sequence of Boolean operations on digital functions;
- Describe each operation in terms of Boolean expression (i.e., a combination of logic operators AND, OR, and NOT);
- Implement the Boolean expression with binary logic gates made from MOS FETs.

This conventional procedure has become the mainstream in information processing. However, there are many potential ways of processing that are different from the conventional procedure. Let us call these processing ways unconventional computing. Unconventional computing is much more sophisticated in processing than the conventional procedure; consequently, it is more powerful for implementing a given digital function with a smaller number of logic gates. Unfortunately, unconventional computing is not easy to implement on LSIs because the CMOS transistor gate—a device suitable for manipulating Boolean expressions—is the only device we can use at this time for constructing practical LSIs. However, unconventional computing can be expected to begin its rise with the development of nanotechnology and quantum devices because quantum devices fabricated using nanotechnology will provide functional properties that can be well used for implementing unconventional computing. As an example of leading examples, this paper outlines the study on single-electron logic circuits that perform logic operation on the basis of a graphical representation of digital functions called the binary decision diagram.

2. Single-Electron Logic Circuits Based on the Binary Decision Diagram

2.1 Representing Digital Functions by Binary Decision Diagrams

The binary decision diagram (called BDD) is a way of representing digital functions by using a directed graph instead of a Boolean expression. It can represent any digital function and provides a concise representation for most digital functions encountered in logic-design applications (Akers [1] and Bryant [2] have given details).

As an example, consider a four-variable digital function represented by the Boolean equation given in Fig. 1(a), This function can also be represented by a BDD shown in Fig. 1(b). A BDD is a graph consisting of many nodes and two terminals, with each node labeled by a variable; in this example, each node is represented by a circle labeled by a variable X_i (i = 1, 2, 3, 4), and a terminal is represented by a square labeled "0" or "1". Each node in a BDD has two branches labeled 1 and 0 and is connected to its adjacent nodes with the branches.

In determining the value of the function for a given set

Manuscript received July 7, 2006.

[†]The author is with the Department of Electrical Engineering, Hokkaido University, Sapporo-shi, 060-0814 Japan.

a) E-mail: amemiya@sapiens-ei.eng.hokudai.ac.jp

DOI: 10.1093/ietele/e89–c.11.1504

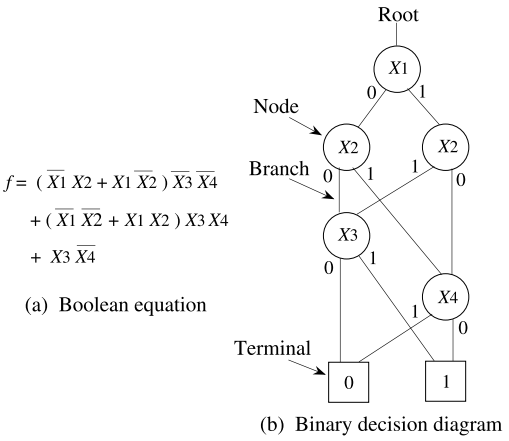


Fig. 1 Graphical representation of digital functions.

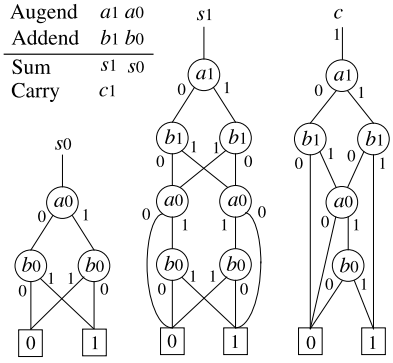


Fig. 2 A set of BDD graphs representing 2-bit addition.

of variable values, we enter at the root and proceed downwards to a terminal. At each node, we follow the branch corresponding to the value of the variable; that is, we follow the 1 branch if $X_i = 1$ and the 0 branch if $X_i = 0$. For a given set of variable values, there is one and only one path from the root to either terminal. The value of the function is equal to the value of the terminal we reach; the function is 1 if we reach the 1-terminal and 0 for the 0-terminal.

As another example of the BDD, the operation of a 2-bit adder is illustrated in Fig. 2. The adder accepts two 2-bit adder inputs (an addend and an augand) and produces the corresponding 2-bit sum output and 1-bit carry output. The two adder inputs is represented by binary numbers a_1a_0 and b_1b_0, the adder output by s_1s_0, and the carry output by c_1, where the value of each element a_1 through c_1 is either 1 or 0. Each bit of the outputs is produced by the corresponding BDD graph, which contains the value of the input bits (a_1, a_0, b_1, and b_0) as node variables.

2.2 Implementing a Binary Decision Diagram with Single-Electron Devices

As described in the previous section, when an input (a set of variable values) is presented to a BDD, a path through the BDD can be traced from the root to either terminal. In this condition, a signal that is injected in the root can travel along the path to reach a 1- or 0-terminal. Therefore, we can determine the value of the function by observing which terminal the signal reaches. This signal is called a messenger.

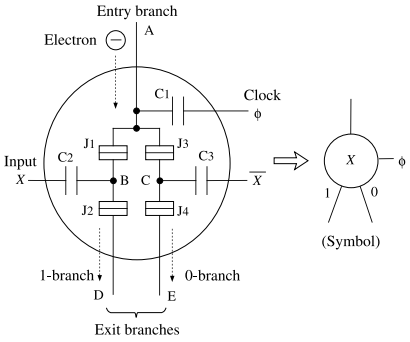


Fig. 3 Unit element (BDD device) for single-electron BDD logic circuits. Its function is to provide two-way switching for electron transport (Asahi, et al. [4]).

A BDD is composed of many identical interconnected nodes, so the node is the unit element of a BDD. The function of the element is simple two-way switching controlled by an input variable. To implement this function, we can use many physical effects that change the course of a traveling messenger (electron, photon, single flux quantum, etc.) in response to an input. The advantage of BDD logic systems is that they can make use of even a physical effect that is useless for constructing transistor-like devices.

Asahi and others proposed implementing the BDD node function by means of single-electron circuits [3]–[5]. Their unit element, a BDD device, illustrated in Fig. 3, consists of four tunneling junctions (J1-J4) and three capacitors (C1-C3) and is driven by a voltage clock (ϕ). It has entry branch (A) and two exit branches (D, E). Voltage input X (and its complement $\overline{X}$), specifying the value of a variable, is applied to island B (and C) through capacitor C2 (and C3); X is an appropriate positive voltage (and $\overline{X}$ is a negative one) if variable = 1, and X is negative ($\overline{X}$ is positive) if variable = 0.

The node device receives a messenger electron from a preceding device through the entry branch and sends the electron to a following device through either exit branch that corresponds to the binary value of the input. The path of the electron transport is A→B→D (the 1 branch) if input X is positive, and A→C→E (the 0 branch) if input X is negative.

2.3 Constructing BDD Logic Circuits

A logic circuit can be constructed by connecting many BDD devices in a cascade manner to build the tree of a BDD graph, as illustrated in Fig. 4. Each BDD device corresponds to a node of the graph and operate as a two-way switch for the transport of a messenger electron. As an example, this section shows a single-electron BDD circuits that implement the 2-bit adder operation represented by the BDD graph given in Fig. 2.

To illustrate the single-electron circuits that implement the BDD graphs, Fig. 5 defines symbols for two devices —

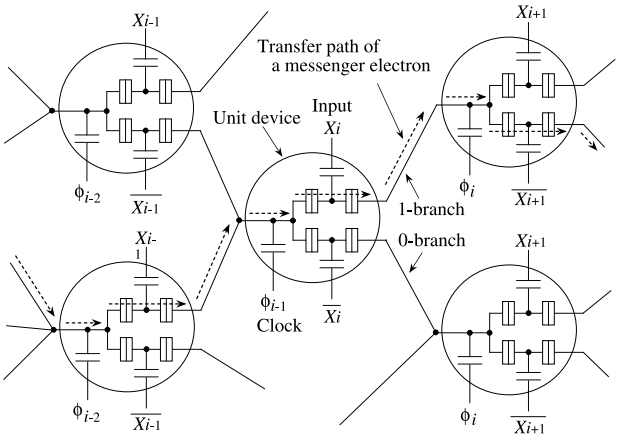


Fig. 4 BDD devices cascaded to build the tree of a BDD graph. Dashed lines represent a path of a messenger-electron transfer (Asahi, et al. [4]).

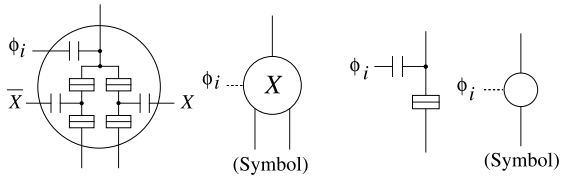


Fig. 5 Symbols of the BDD device and the buffer.

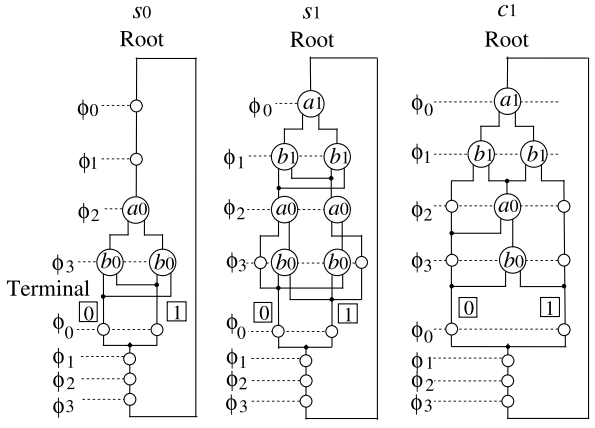


Fig. 6 Circuit configuration of the 2-bit adder (Asahi, et al. [5]).

the BDD device and a buffer. The buffer is a subcircuit consisting of a tunneling junction and a capacitor. Its function is to set up a dummy node and holds a messenger electron for one clock period — an indispensable timing element for constructing single-electron BDD circuits.

The 2-bit adder designed by Asahi and others are illustrated in Fig. 6. In each circuit, two terminal nodes are indicated by the boxed numerals 1 and 0. The configuration of each circuit was obtained by simply replacing the nodes in the BDD graphs of Fig. 2 with BDD devices. Factors to keep in mind in designing the circuit include the following:

- To transfer and circulate the messenger electron in the circuit, the four-phase clock (ϕ_0-ϕ_3) is applied to node devices and buffers. The phase shift of the clock is: $\phi_0 = 0$, $\phi_1 = -\pi/2$, $\phi_2 = -\pi$, $\phi_3 = -3\pi/2$. The bit signals of the adder inputs are applied in sequence to the BDD devices such that the bit signal for a BDD device is applied synchronously with the clock pulse for the consecutive BDD device (or the consecutive buffer);
- In actual systems, an input-bit signal will be applied to all the BDD circuits simultaneously with the clock. For successful operation in such a situation, the buffers (dummy nodes) have to be set up on appropriate points on the path to ensure that a messenger electron will arrive at each BDD device at the correct time;
- Two or more messenger electron can be used in a BDD circuit. Putting a messenger electron on every transfer stage (a subcircuit that is driven by a set of four clock pulses) produces a pipelined operation that improves the processing throughput. In the present instance, two messenger electrons can be put in each BDD circuit.

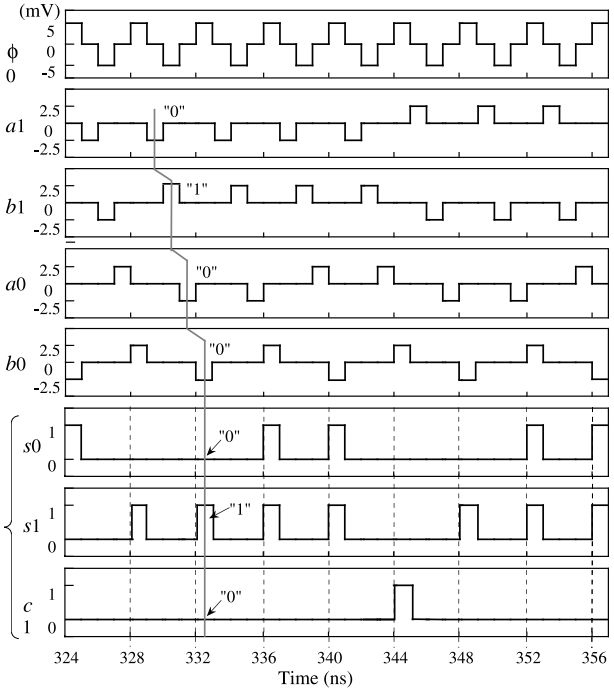


Fig. 7 Operation of the 2-bit BDD adder (simulation). A set of input and output data is marked by a gray line (Asahi, et al. [5]).

2.4 Logic Operation of BDD Circuits

Asahi and others confirmed the operation of their BDD adder by computer simulation [4], [5]. Figure 7 shows part of their results. The parameters used in this example are: junction capacitance = 10 aF and tunnel resistance = 100 kΩ for each tunnel junction, capacitance = 10 aF for each capacitor, and temperature = 0 K. Two messenger electrons were put in each BDD circuit. The figure plots the waveforms for clock pulse ϕ_0 (the other clocks are omitted), input-bit pulses (a_1, a_0, b_1, and b_0), and output charges (s_1, s_0, and

c_1, the charges on the 1 terminal nodes of the three BDD circuits, normalized to the electron charge). The adder produces an output data flow in response to the input data flow; in the figure, a set of input and output data is marked by a gray line.

3. Logic Circuits Based on the Shared BDD

As described in Sect. 2, a BDD has a path from the root to either terminal for a given set of variable values. Therefore, we can also determine the value of the digital function by tracing the path upward from the 1 terminal to the root to check whether the 1-terminal is connected to the root. If the connection is established, the function is logical 1, and if not, the function is logical 0. This upward tracing leads us to another form of single-electron BDD circuits, as shown in the following subsections.

In the upward tracing, each node in BDD acts as a switching element with two entry branches (1-branch and 0-branch) and an exit, as shown in the left side in Fig. 8. In each node in BDD graphs, one entry branch corresponding to the value of the variable is connected to the exit and the other branch is disconnected; e.g., if the variable is logical 1, the 1-branch is connected to the exit and the 0-branch is not connected. To determine the value of the logic function, we check whether the 1-terminal is connected to the root; if the connection is established, the function is logical 1, and if not, the function is logical 0. Most digital systems contain many functions that are closely related to one another, and these functions can be represented by a single graph with multiple roots (one root for each function). Figure 8 shows an example, a representation of 2-bit addition with three roots for three output bits (a 2-bit sum output and a 1-bit carry output). This BDD graph is a combination of the three graphs given in Fig. 2. This kind of BDD is called a shared BDD.

3.1 Constructing Shared-BDD Circuits Combined with the Upward Tracing

Yamada and others proposed implementing shared BDDs by means of tunneling gates [6]. The node function for the upward tracing can be implemented with tunneling gates. The tunneling gate is a tunneling junction with a gate electrode that controls electron transport through the tunneling junction (Fig. 9(a)). It accepts a binary gate voltage as input, and transmits electrons through the junction if the gate voltage is logical 1 (the junction is on), and transmits no electrons if the gate voltage is logical 0 (the junction is off). The tunneling gates can be made with quantum-dot devices as shown later in Fig. 13.

Figure 9(b) shows the BDD device consisting of two tunneling gates and a ground capacitor joined together at the exit node. A binary gate voltage (and its complement), specifying the value of an input variable x_i, is applied to tunneling gate labeled x_i (and gate labeled $\overline{x_i}$) to control the tunneling transport of the junction. If variable $x_i = 1$ ($\overline{x_i} = 0$), the BDD device transports electrons from the 1-branch to the exit, and if $x_i = 0$, it transports from the 0-branch to the exit.

Any combinational logic can be implemented by combining the BDD devices to build a BDD graph circuit. In operating the circuit, we apply input gate voltages to the tunneling gates and inject electrons into the circuit from the 1-terminal, and then observe at each root whether the electrons flow out or not. As an example, Fig. 10 shows a 2-bit adder circuit that implements the shared BDD given in Fig. 8. The circuit is designed on the basis of the following principles.

- In composing the circuit, the 0-terminal and related branch connections in BDDs are unnecessary and removed.
- A gradient potential from the 1-terminal to the roots must be appropriately established so that, in every BDD device, electrons can flow from the entry branches to the exit. This can be achieved by inserting dummy tunneling junctions in electron paths so that every path from the 1-terminal to a root will have the same number of tunneling junctions.

The circuit accepts two 2-bit binary inputs (augend a_1a_0 and addend b_1b_0) and produces the corresponding 2-bit sum output (s_1s_0) and a one-bit carry output (c_1). The input voltages are applied to the tunneling gates labeled with variables a_1 through b_0 to control (turn on and off) electron

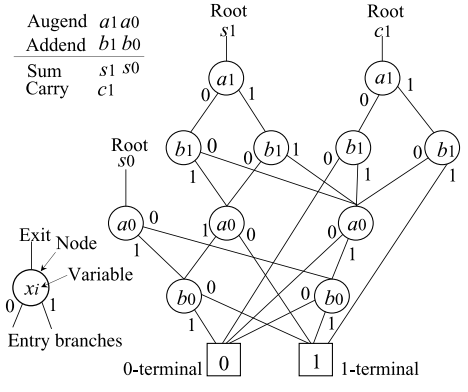


Fig. 8 Shared-BDD representation for 2-bit addition.

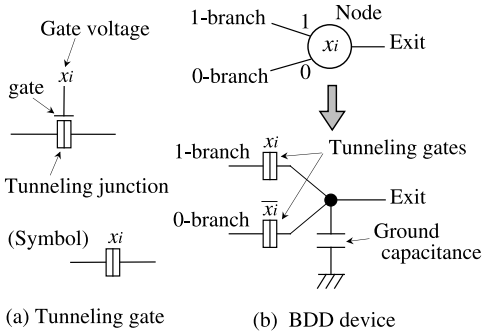


Fig. 9 BDD device consisting of tunneling gates (Yamada, et al. [6]).

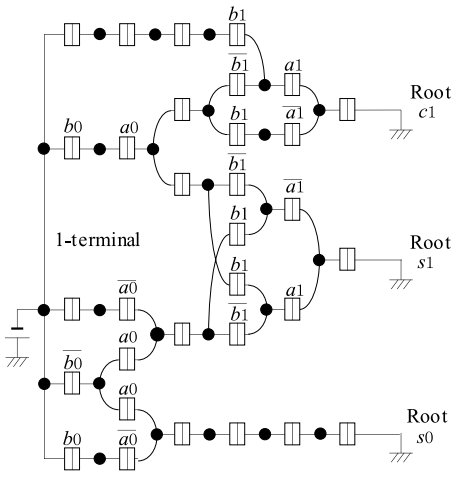


Fig. 10 2-bit adder circuit implementing the shared BDD graph in Fig. 8. The ground capacitance for each node is omitted (Yamada, et al. [6]).

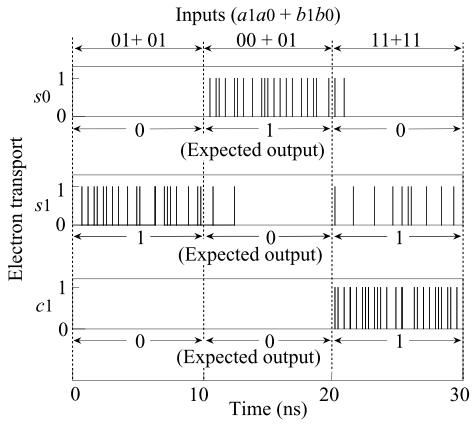


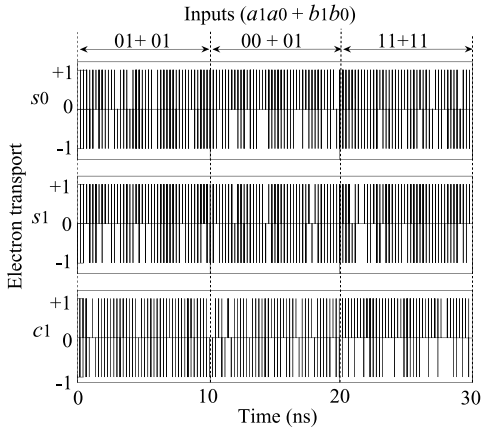
Fig. 11 Operation of the adder circuit in Fig. 13 (simulation). Each impulse signifies the arrival of an electron at a root. The expected logical values for each output bit are written in the figure. Temperatures is assumed to be 0 K (Yamada, et al. [6]).

transport through the tunneling gates. The nonlabeled junctions are dummy tunneling junctions that establish a potential gradient through electron paths. Each node has a ground capacitance, but for simplicity the capacitance is omitted in the figure.

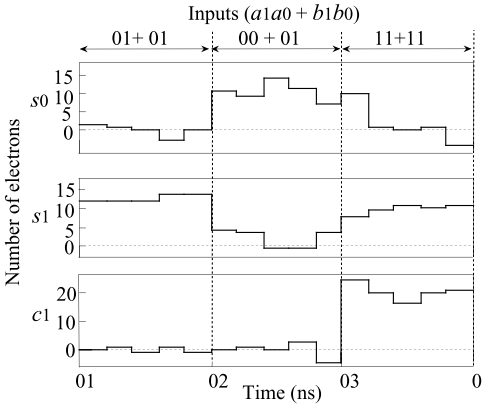
To operate the circuit, all the roots are grounded and a negative power voltage is applied to the 1-terminal. Electrons are injected from the power voltage into the circuit and transported toward each root along the paths specified by the variables. The logical value of an output is 1 if electrons can reach the corresponding root, and 0 if they cannot. The circuit is analogous in operation to pass-transistor circuits composed of MOSFETs, and it can be considered to be an ultra-low-power version of a pass-transistor circuit. In the circuit, the electron flow (therefore power consumption) is regulated by the Coulomb blockade and can be set to a far smaller quantity than can be in ordinary pass-transistor circuits. The number of electrons that flow through the circuit during one logic operation can be reduced to tens or so by adjusting the tunnel junction and the ground capacitances (see Fig. 11).

3.2 Logic Operation of a 2-bit Adder Consisting of a Shared-BDD Circuit

Yamada and others confirmed the add operation of their circuits by computer simulation [6]. Figures 11 and 12 show part of their results for the 2-bit adder. The device parameters used are: tunneling gate junction capacitance = 10 aF, dummy tunneling junction capacitance = 10 aF, ground capacitance of a node = 20 aF, and tunneling gate resistance = 1 MΩ in the 'on' state and 10 GΩ in the 'off' state. The voltage of the 1-terminal was set to −10 mV for the results shown in Fig. 11 and −20 mV for Fig. 12) Figure 11 shows the results for zero temperature (0 K). In the figure, three input combinations (a_1a_0+b_1b_0 = 01+01, 00+01, and 11+11) are applied in sequence to the gates of the circuit at 10-ns intervals. Each impulse in the figure signifies the arrival of



(a) Transport of electrons



(b) Number of output electrons arriving at the root

Fig. 12 Operation of the adder circuit at 20K (simulation) (Yamada, et al. [6]).

an electron at roots s_0, s_1, and c_1. A total of ten electrons flowed out of root s_1 during the period from 20 ns to 30 ns. The circuit produces the expected correct outputs. The number of output electrons, therefore the power consumption of

the circuit, can be controlled by adjusting the 1-terminal.

Figure 12(a) shows the results at a temperature of 20 K. The circuit produces noisy outputs due to thermal agitation; i.e., a countercurrent or countertransport of electrons from a root into the circuit is frequently observed, shown by negative impulses in the figure. Nevertheless, the correct outputs can be retrieved by counting the net number of output electrons. The results are shown in Fig. 12(b). In this example, the counting was repeated every two nanoseconds; e.g., a total of nine electrons flowed out of root s_0 during the period from 12 ns to 14 ns. Using a longer period for the counting can operate the circuit at higher temperatures.

4. Shared-BDD Integrated Circuits Constructed Using a Semoconductor Process Technology

Kasai and others developed single-electron subsystems based on the shared BDD, with the aim of constructing a microprocessor [7], [8]. To make BDD devices, they used quantum-dot arrays formed in the wrap-gate structure (WPG). The WPG is a quantum wire with two or more metal gates that control electron transport along the wire [9], [10], as shown in Fig. 13(a). It consists of a quantum wire of two-dimensional electron gas (2DEG) in a GaAs-GaAlAs heterostructure, with Schottky gates wrapped around the quantum wire. In this suructure, depletion layers or potential barriers extend from the gates into the quantum wire. These potential barriers divide the quantum wire into many separate dots, forming a quantum-dot array (Fig. 13(b)). In this array, electrons can be transported by tunneling through the potential barrier from one dot to adjacent dots.

Electron transport between the adjacent two dots can be modulated by controlling the voltage of the gate to regulate the width of the potential barrier. Electrons will be transferred by tunneling from one dot to another if the gate is set to a low negative voltage, while no electrons will be transferred if the gate is set to high negative voltage. A two-way switch required for BDD devices can be implemented using a forked quantum wire with two gates (gate 1 and gate 0) wrapped around the wire, as shown in Fig. 13(c). A binary voltage input (and its complement), specifying the value of a variable, is applied to gate 1 (and gate 0) to regulate the resistance of the tunnel junctions under the gates; e.g., if variable x_i is logical 1, a low negative voltage is applied to gate 1 and a high negative voltage to gate 0. The device enables electron transport by tunneling from the 1-branch to the exit if $x_i = 1$ ($\overline{x_i} = 0$), and from the 0-branch to the exit if $x_i = 0$ ($\overline{x_i} = 1$).

To construct integrated circuits with the BDD devices,

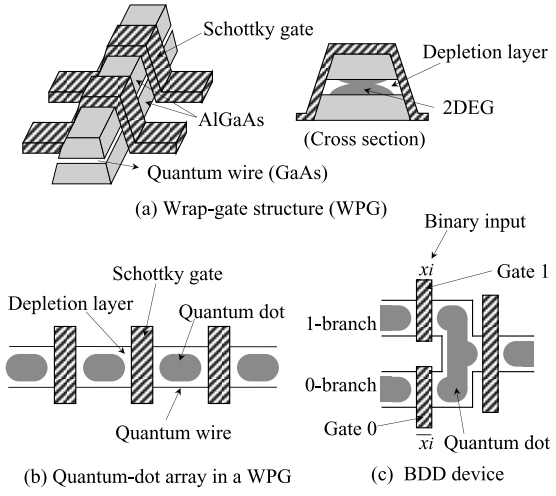


Fig. 13 BDD device consisting of quantum-dot tunneling gates: (a) wrap-gate structure, (b) quantum-dot array, and (c) BDD device (Kasai, et al. [9] and Yamada, et al. [6]).

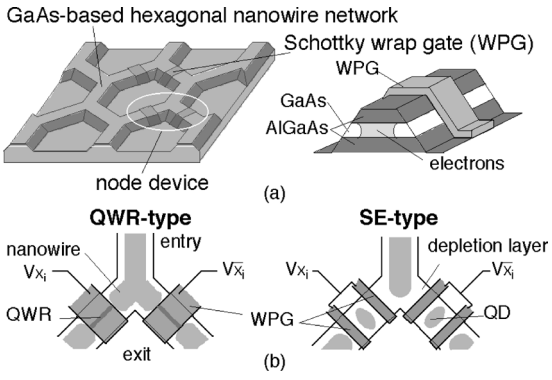


Fig. 14 BDD devise constructed with Hexagon NanoProcess: (a) hexagonal network of quantum wires and a tunneling gate with a Schottky gate (WPG), and (b) two structures of BDD devices, i.e., a quantum-wire (QWR) device consisting of two tunneling gates and a single-electron transistor (SE) device consisting of four tunneling gates (Kasai, et al. [8]).

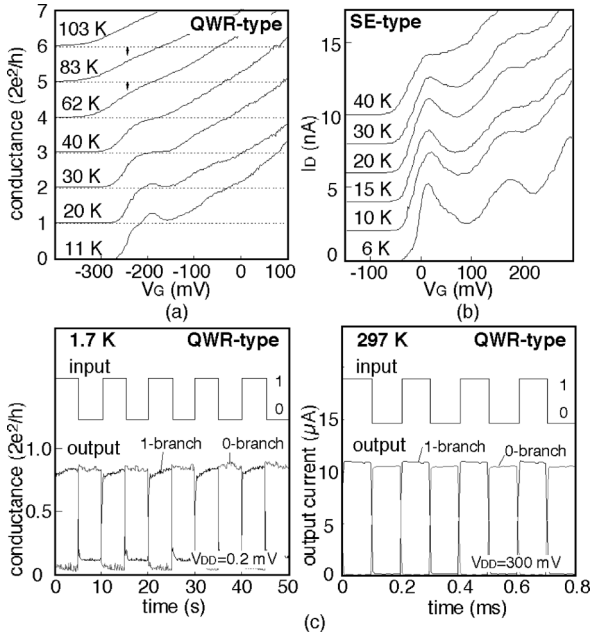


Fig. 15 On-off conductance as a function of gate voltage for (a) tunneling gate and (b) single-electron-transistor switch consisting of the tunneling gates, measured at various temperatures; and (c) two-way switching of the BDD device consisting of the tunneling gates, measured at 1.7 K and 297 K (Kasai, et al. [8]).

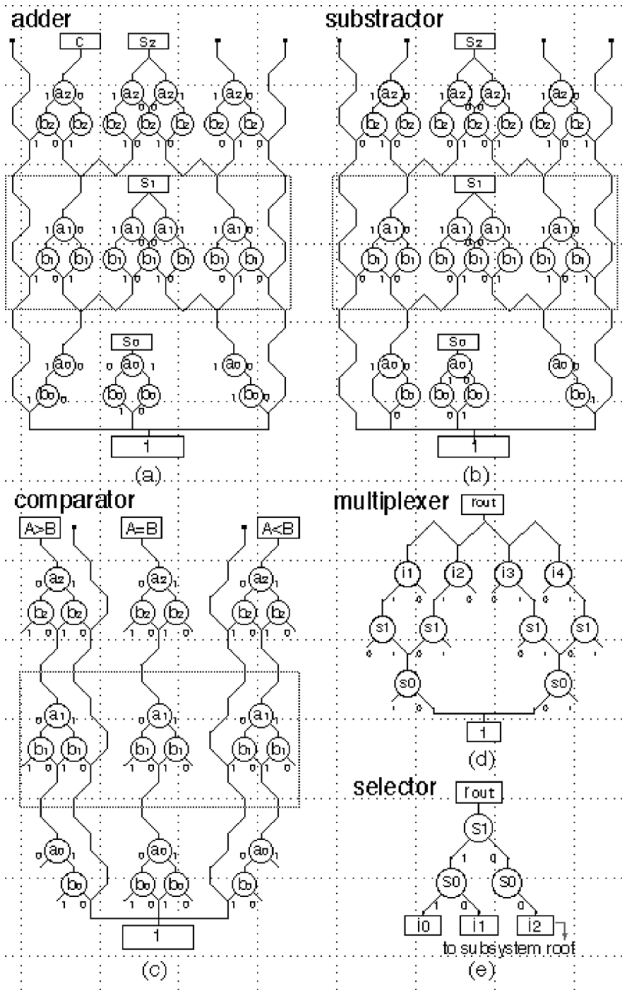


Fig. 16 Shared-BDD graphs designed to construct subsystems for a single electron 2-bit processor (Kasai, et al. [8]): (a) adder, (b) subtractor, (c) comparator, (d) multiplexer, and (e) selector.

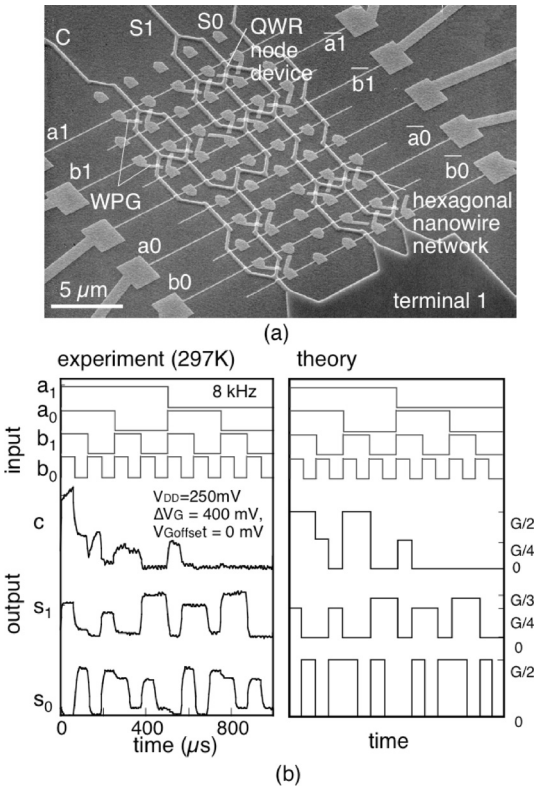


Fig. 17 Single electron 2-bit adder fabricated with Hexagon NanoProcess technology: (a) SEM image of the device and (b) input-output waveforms measured at room temperature (left figure) compared with simulation (right figure) (Kasai, et al. [8]).

Kasai and others used a sophisticated process technology called Hexagon NanoProcess [7], [8]. In the first step of their process, a hexagonal network of GaAlAs-GaAs heterostructure nanowires is formed on an insulating GaAs layer (the left figure in Fig. 14(a)). A quantum wire that consists of two-dimensional electron gas (2DEG) goes through the nanowire (the right in Fig. 14(a)). In the second step, a Schottky gate consisting of metal is formed on the wire to construct a tunneling gate (the right in Fig. 14(b)). The Schottky gate is wrapped around the wire, and a depletion layer or a potential barrier extends from the gate into the quantum wire. This potential barrier divides the quantum wire into two separate 2DEG regions, and a tunneling junction is formed between the two 2DEG regions. With these tunneling gates, BDD devices can be constructed as shown in Fig. 14(b). The left device consists of two tunneling gates, and the right device uses two single-electron transistor switches consisting of four tunneling gates. They were fabricated on a GaAlAs-GaAs hexagonal nanowire network that was formed by chemically etching a AlGaAs/GaAs heterostructure layer on an insulating GaAs substrate. The Schottky gate were attached on the nanowires by using EB lithography, metal deposition, and lift-off process. The width of the nanowire was 300–500 nm, and the gate length was 300 nm. The on-off conductance and the two-way switching of the BDD devices are shown in Figs. 15(a) and 15(b).

With these BDD devices, Kasai and others constructed several subsystems for a single-electron 2-bit processor [8]. Figure 16 shows the shared BDD graphs for this purpose. To implement circuits on a hexagonal network on a substrate, a BDD graph for each subsystem was revised so that it would have no intersection of branches; e.g., the adder graph given in Fig. 8 was revised into the graph shown in Fig. 16(a). As an example of fabricated subsystems, a 2-bit adder is shown in Fig. 17; the figure (a) shows a SEM image, and the figure (b) shows input-output waveforms of 2-bit add operation for all possible input combinations. Using Hexagon NanoProcess will enable us to construct single-electron LSIs with large scale integration of 25–45 million BDD devices/cm^2.

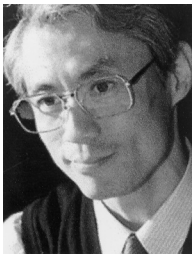
5. Conclusion

Unconventional computing is a promising way of performing functional, powerful processing but is not easy to implement on LSIs with MOS FET gates. However, uncon-

ventional computing can be expected to begin its rise with the development of nanotechnology and quantum devices because quantum devices fabricated using nanotechnology will provide functional properties that can be used for implementing unconventional computing. As an example of leading examples, this paper outlined the single-electron logic circuit based on the BDD. With single-electron BDD circuits, we will be able to integrate large-scaled digital circuits on a chip and proceed from there to develop unconventional-computing LSIs.

References

[1] S.B. Akers, "Binary decision diagrams," IEEE Trans. Comput., vol.C-27, no.6, pp.509–516, 1978.
[2] R.E. Bryant, "Graph-based algorithms for Boolean function manipulation," IEEE Trans. Comput., vol.C-35, no.8, pp.677–691, 1986.
[3] N. Asahi, M. Akazawa, and Y. Amemiya, "Binary-decision-diagram device," IEEE Trans. Electron Devices, vol.42, no.11, pp.1999–2003, 1995.
[4] N. Asahi, M. Akazawa, and Y. Amemiya, "Single-electron logic device based on the binary decision diagram," IEEE Trans. Electron Devices, vol.44, no.7, pp.1109–1116, 1997.
[5] 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, Jan. 1998.
[6] T. Yamada, Y. Kinoshita, S. Kasai, H. Hasegawa, and Y. Amemiya, "Quantum-dot logic circuits based on the shared binary decision diagram," Jpn. J. Appl. Phys., vol.40, pp.4485–4488, 2001.
[7] S. Kasai, M. Yumoto, T. Tamura, and H. Hasegawa, "Feasibility study of GaAs digital processors based on hexagonal BDD quantum logic circuit approach," Conference Digest of 61st Device Research Conference (DRC), pp.97–98, 2003.
[8] S. Kasai, M. Yumoto, T. Tamura, and H. Hasegawa, "Design and implantation of hexagonal quantum BDD logic subsystems for nanoprocessor arithmetic logic processing," Proc. 2003 Asia-Pacific Workshop on Fundamentals and Application of Advanced Semiconductor Devices, pp.177–180, 2003.
[9] S. Kasai, K. Jinushi, H. Tomozawa, and H. Hasegawa, "Fabrication and characterization of GaAs single electron devices having single and multiple dots based on Schottky in-plane-gate and wrap-gate control of two-dimensional electron gas," Jpn. J. Appl. Phys., vol.36, pp.1678–1685, 1997.
[10] Y. Satoh, S. Kasai, K. Jinushi, and H. Hasegawa, "Computer simulation and experimental characterization of single electron transistors based on Schottky wrap gate control of 2DEG," Jpn. J. Appl. Phys., vol.37, pp.1584–1590, 1998.



Yoshihito Amemiya received the B.E., M.E., and Dr. Engineering degrees from the Tokyo Institute of Technology, Tokyo, Japan, in 1970, 1972, and 1975. He joined NTT Musashino Laboratories in 1975, where he worked on the development of silicon process technologies for high-speed logic LSIs. From 1983 to 1993, he was with NTT Atsugi Laboratories and developed bipolar and CMOS circuits for Boolean and non-Boolean logic LSIs. Since 1993, he has been a Professor with the Department of Electrical Engineering, Hokkaido University, Sapporo. His research interests are in the fields of silicon LSI circuits, signal processing devices based on nonlinear analog computation, logic systems consisting of single-electron circuits, and information-processing devices making use of quantum nanostructures.