IEICE TRANS. ELECTRON., VOL. E81–C, NO. 1 JANUARY 1998

PAPER Special Issue on Technology Challenges for Single Electron Devices

Single-Electron Majority Logic Circuits

Hiroki IWAMURA[†a)], *Student Member*, Masamichi AKAZAWA[†], *and* Yoshihito AMEMIYA[†], *Members*

SUMMARY This paper proposes an architecture for circuit construction for developing single-electron integrated circuits based on majority logic. The majority logic gate circuit proposed consists of a capacitor array for input summation and a single-electron inverter for threshold operation. It accepts an odd number of inputs and produces the corresponding output on the basis of the principle of majority decision; it produces an output of logic "1" if the majority of the inputs is 1, and an output of "0" if the majority is 0. By combining the proposed majority gate circuits, various subsystems can be constructed with a smaller number of devices than that of Boolean-based construction. An adder and a parity generator are designed as examples. It is shown by computer simulation that the designed subsystems produce the correct logic operations. The operation error induced by thermal agitation is also estimated.

key words: *single electron majority logic, threshold logic, gate, circuit, thermal noise*

1. Introduction

One of the promising areas of research in microelectronics is the development of integrated circuits based on single-carrier electronics. For this purpose, we must create a single-electron logic device that can be used in constructing large digital processing systems. This paper proposes such a new logic device: namely, a *single-electron majority logic circuit.*

Single-carrier electronics is a technology for manipulating electronic functions by controlling the transport of individual electrons, utilizing the single-electron tunneling phenomenon. It has been receiving increasing attention because it affords the possibility of producing LSIs that combine large integration and low power dissipation. To take steps towards this goal, we must develop digital devices that can perform complex and high-level logic operations through the use of single electrons. In this paper, we will propose constructing such a device: that is, a single-electron digital circuit that is based on the concept of *majority logic.*

The majority logic method is a way of implementing digital operations in a manner different from that of Boolean logic. Instead of using Boolean logic operators (AND, OR, and their complements), majority logic represents and manipulates digital functions on the basis of the principle of *majority decision.* The logic process of majority logic is much more sophisticated than that of Boolean logics; consequently, majority logic is more powerful for implementing a given digital function with a smaller number of devices.

The prospects for future applications of majority logic are wholly dependent on whether a logic device is feasible that is suitable for majority logic operation. In the late 1950's, several computer systems based on the majority logic architecture were developed and constructed for practical use, by using a functional device called the *parametron* (a majority logic device that utilizes the phenomenon of parametric phase-locked oscillation). After that, however, majority logic had to leave the stage, because the CMOS circuit — a natural Boolean logic device — came to be a dominant device in electronics. But majority logic can be expected to make a comeback with the advent of single-carrier electronics because, as described later, the single-electron circuit will provide excellent and elegant devices for majority logic systems.

In the following sections, we propose that majority logic is a suitable architecture for constructing digital logic systems from single-electron circuits. We will first propose a gate-circuit structure for implementing the unit function required for majority logic operation. Then we will discuss the issues that require attention in determining the parameters of the proposed gate circuit (Sect. 2). After that we will design two sample subsystems (an adder and a parity generator) by combining the gate circuits, and will show by computer simulation that the designed subsystems produce the correct operations (Sect. 3). The operation error induced by thermal agitation will be estimated (Sect. 4). Finally, we will conclude by suggesting future development in single-electron majority logic systems.

2. Implementing Majority Logic by Using Single-Electron Circuits

2.1 Unit Function of Majority Logic

The basic operation or unit function of majority logic is *to decide the output state by a majority vote of input states.* This function is illustrated in Fig. 1. The logic element (a majority gate) has an *odd* number of binary

Manuscript received June 18, 1997.
Manuscript revised September 16, 1997.

[†]The authors are with the Faculty of Engineering, Hokkaido University, Sapporo-shi, 060 Japan.
a) E-mail: iwamura@sapiens.huee.hokudai.ac.jp

inputs and a binary output (a five-input gate is shown in Fig. 1 (a)). It produces an output of 1 if the majority of the inputs is 1, and produces an output of 0 if the majority is 0 (solid curve in Fig. 1 (b)); e.g., if the five inputs are 1, 0, 0, 1, and 1, then the output is 1, and if the inputs are 1, 0, 1, 0, and 0, then the output is 0. Any digital functions can be implemented by a combination of the majority gates and inverters. For details of majority logic, see Refs. [1] and [2]. A three-input gate suffices for designing any logic system, but if a larger input gate is available, then concise construction of logic systems will be practicable. In general, three-input and five-input gates are used.

Illustrated in Fig. 1 (b) by a dashed curve is the *complementary majority function* (the complement of the unit function above). Every digital function can be implemented by using *only* the gates of the complementary majority function, but it is advisable to use inverters jointly for designing logic systems concisely.

The majority logic provides a complete and concise implementation for most functions encountered in logic design applications. As an example, the implementation of a single-bit full adder is illustrated in Fig. 2. In this construction, the *complementary majority-function gates* are used because the gate circuit that will be proposed in the next section produces the complementary function. (In the figure, an inverter is represented by a segment on a connection branch, according to precedents in the flow-diagram description for majority logic.) As can be seen, a full adder is composed of only three gates with two inverters. In contrast, the Boolean-based implementation requires a larger circuit construction corresponding to seven or eight gate elements.

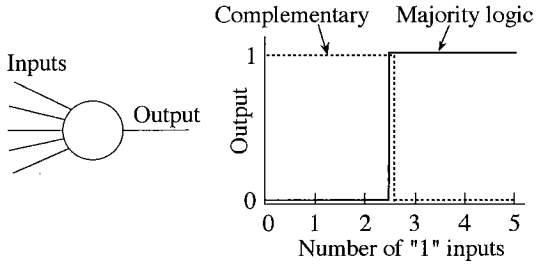


Fig. 1 Majority gate: (a) a symbol for a majority gate, and (b) the unit function of majority logic. The solid curve is the unit function of majority logic, and the dashed curve is the complementary function. An instance for a five-input gate is illustrated.

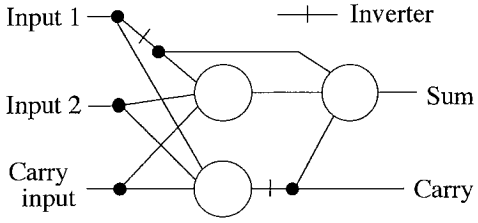


Fig. 2 Single-bit full adder composed of three majority gates and two inverters. In this construction, each symbol denotes a complementary function gate. An inverter is represented by a segment on a connection branch.

2.2 Constructing a Majority Gate Using Single-Electron Circuits

The essential function for the majority gate operation is to ascertain whether the majority of inputs is 1 or 0. This can be performed by calculating the mean of the inputs (each input is 1 or 0) and comparing the mean value with 0.5; if the mean value is larger than 0.5, the majority can be considered to be 1, and if smaller than 0.5, then the majority can be considered to be 0. One way of implementing this function into an electronic circuit is to use the technique of Kirchhoff resistor summation or that of charge-balancing capacitor summation, with a threshold device for producing the corresponding 1-0 output. Because the single-electron circuit is an electronic circuit consisting only of tunnel junctions and capacitors, it is natural to use the capacitor summation technique to produce the mean value of inputs.

The majority gate circuit we propose is illustrated in Fig. 3 (a), with an example of a three-input configuration. It consists of an input capacitor array (six capacitors C) for input summation and an inverter subcircuit (tunnel junctions C_{j1} through C_{j4} and capacitors C_1 through C_3) for threshold operation. The circuit accepts three input voltages V_1, V_2, and V_3 and produces the corresponding output voltage V_{out}. The configuration of this circuit is modeled on the single-electron inverter proposed by Tucker, [3] illustrated in Fig. 3 (b); that is, starting with the inverter configuration of Fig. 3 (b), we divided each of the two input capacitances (C_0 in Fig. 3 (b)) into three equal capacitances, then connected the divided capacitances (C in Fig. 3 (a)) to three input terminals to create the majority gate. The gate input

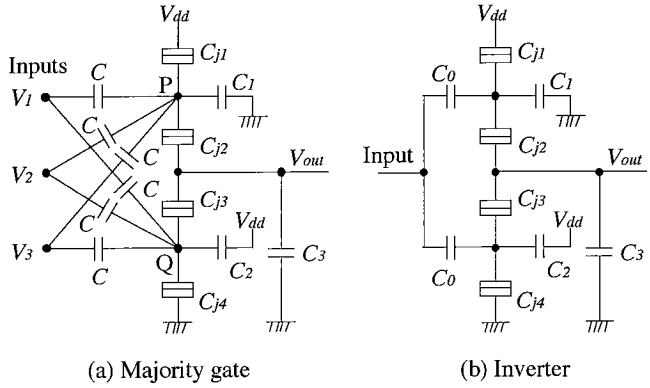


Fig. 3 Single-electron majority gate circuit: (a) the circuit configuration of the majority logic gate, and (b) the Tucker-type inverter circuit.

capacitance C is set at *one third* of the inverter input capacitance C_0, that is, $C = C_0/3$ (other parameters are set to be the same as those of the original inverter). If an N-input gate is needed, $2N$ input capacitors will be used. The capacitance of each of these is set at $1/N$ of the inverter input capacitance (i.e., $C = C_0/N$).

This gate circuit produces the complementary majority function as follows. The input nodes P and Q of the inverter subcircuit are coupled to each input V_1, V_2, or V_3 through each input capacitance C, so the potential of each input node is changed in proportion to the mean value of the inputs. Therefore, if two or three inputs are 1, the gate circuit is in the same state as that of an inverter that is applied an input of larger than 0.5, and so the corresponding output will be 0. And if two or three inputs are 0, the gate state is the same as the inverter state for an input of smaller than 0.5, so the output will be 1.

(In actual device construction, a number of parasitic capacitances will be added to the circuit configuration of Fig. 3 (a). But for simplicity, we will ignore them in the following.)

2.3 Determining the Gate-Circuit Parameters

To achieve the correct gate operation, consideration must be given to the stability of the inverter subcircuit: that is, the inverter subcircuit has to be designed so that it will be stable throughout its transfer curve.

As is well known, the single-electron circuit has many possible internal states and frequently shows unstable behavior in which the circuit state, under a given input, varies constantly between two or more different states. The unstability can be removed by a careful selection of circuit parameters, but in practice, the unstability has been left as it is unless it impedes binary logic applications. As for the inverter circuit, it has been operated in general under conditions such that it has an unstable region on its transfer curve for intermediate values of the input (Fig. 4 (a)).

This is not a problem for a binary-logic inverter. But it is fatal to a majority gate application, because in the majority gate the inverter subcircuit will frequently receive an intermediate value of the input. In designing the majority gate circuit, it is required that the inverter subcircuit should be stable throughout its transfer curve: that is, the inverter subcircuit has to have the characteristic of a *step inverter* (Fig. 4 (b)).

A set of the circuit parameters for creating the step characteristic can be determined by calculating and scrutinizing the *stability diagram* of the circuit, in the way as described in Ref. [4]. In designing the majority gate circuit in Fig. 3 (a), we first determined the optimum parameter set for the inverter circuit of Fig. 3 (b), then transcribed the determined parameters for the majority gate circuit. Several sets of parameters were discovered. A sample set for a *three*-input majority gate

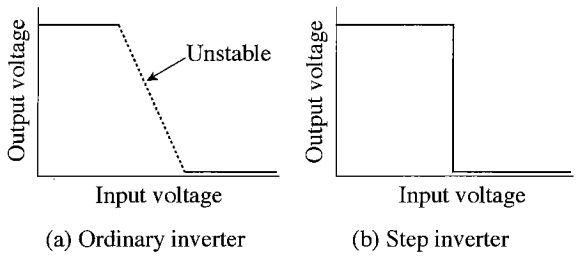


Fig. 4 Transfer curve of a single-electron inverter: (a) an ordinary inverter characteristic, and (b) a step inverter characteristic.

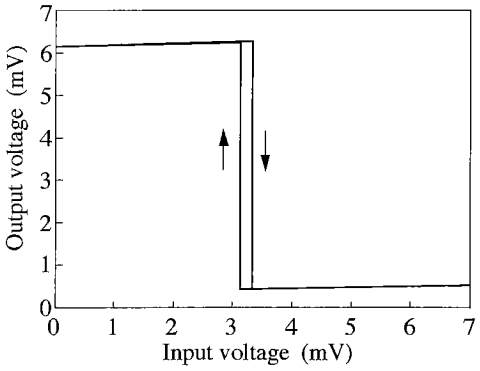


Fig. 5 Transfer curve of the majority gate circuit. Simulated under the condition of all the three input terminals being applied the same input voltage. For the circuit parameters, see the text.

is:
$C = 1$ aF, $C_{j1} = C_{j4} = 1$ aF, $C_{j2} = C_{j3} = 2$ aF,
$C_1 = 9$ aF, $C_2 = 9$ aF, $C_3 = 24$ aF, and $V_{dd} = 6.5$ mV.

To show that the step characteristic is established in the inverter subcircuit, the transfer curve of the gate circuit is illustrated in Fig. 5, simulated under the condition of all the three input terminals being applied the same input voltage. A slight hysteresis is observed at the high-low transition of the transfer curve because the parameters were determined such that the unstability for intermediate inputs would definitely be suppressed. This is not a problem in the present application, and if necessary, the hysteresis can be removed by adjusting the parameters more finely. It is worthy of note that, unlike CMOS inverters, single-electron step inverters do *not* produce short-circuit current even for intermediate values of input. This is quite convenient for low-power operation.

With the same parameters for the inverter subcircuit, we can construct a five-input majority gate by using ten input capacitances, each of which is 0.6 aF instead of 1 aF.

3. Subsystem Design Using the Majority Gate Circuits

Any digital function can be implemented by using the majority gate (Fig. 3 (a)) and the inverter (Fig. 3 (b)).

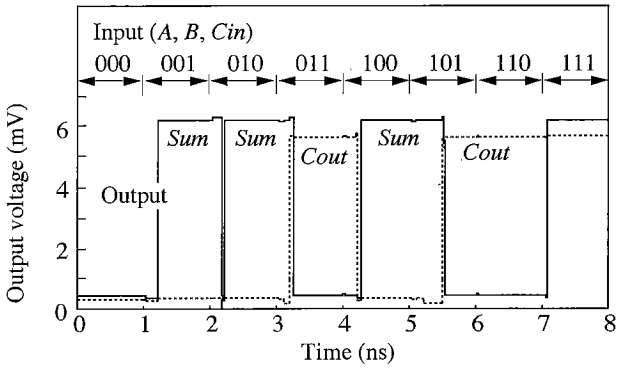


Fig. 6 Add operation of the single-bit full adder (simulation). Plotted are the waveforms of a sum output *Sum* and a carry output *Cout* for the eight input combinations.

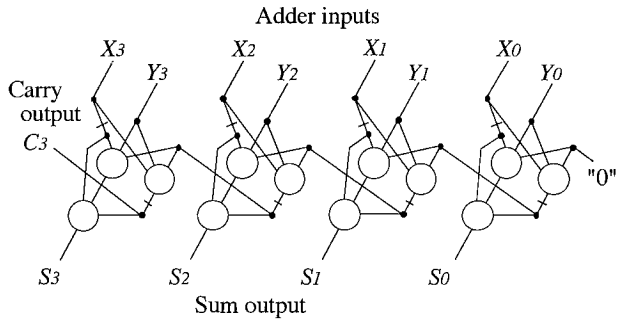


Fig. 7 Construction of a 4-bit ripple carry adder. Each symbol denotes a complementary function gate. An inverter is represented by a segment on a connection branch. This adder calculates the sum of two binary numbers "X3 X2 X1 X0" and "Y3 Y2 Y1 Y0" to produce a sum output "S3 S2 S1 S0" and a carry output "C3."

(This majority gate is a complementary-function gate, and therefore suffices for any logic functions. Our use of the inverters jointly is to simplify circuit construction.) We here present elemental sample designs of single-electron majority subsystems (an adder and a parity generator) and show their logic operation by computer simulation. In the simulation we used the Monte Carlo method combined with the basic equations for electric-charge distribution, charging energy, and tunneling probability (for this method, see Ref. [5]). The parameters used are the same as those given in Sect. 2.3, with a tunnel resistance of 100 kΩ for all the tunnel junctions. The temperature was assumed to be 0K.

3.1 Adder

We first designed the single-bit full adder, illustrated in Fig. 2, using three majority gates and two inverters. A simulated add operation is summarized in Fig. 6. In this figure the three inputs (adder input A, adder input B, carry input Cin) are applied in sequence as (000, 001, 010, 011, 100, 101, 110, 111), and the corresponding two outputs (sum output *Sum*, carry output $Cout$) is observed as (00, 10, 10, 01, 10, 01, 01, 11). The successful add operation can be seen.

Some explication is needed for the operation speed of the circuit. Electron tunneling is in general a probabilistic phenomenon, so the operation delay time in a single-electron circuit is not a fixed value but differs in each operation event. (In simulation, the probabilistic characteristic is taken into account by use of random numbers, and Fig. 6 shows the result for a given set of random numbers.) It is therefore the usual practice for single-electron circuits to represent the operation speed by the mean of the operation delay time. We hereafter follow this practice. In the designed full adder, the mean add time was 0.3 ns.

We then designed a 4-bit ripple carry adder by combining four full adders into a cascade, as illustrated in Fig. 7. It was confirmed by simulation that the adder

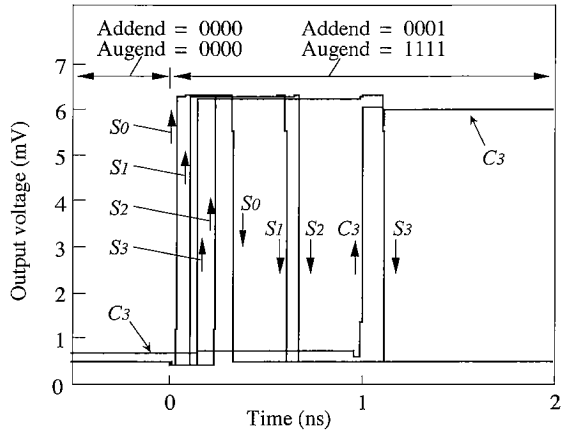


Fig. 8 Operation of the adder in Fig. 7 (simulation). Plotted are the output waveforms (a sum output *Sum* and a carry output *Cout*) for the add operation of 1111 + 0001.

can operate for all input combinations. Illustrated in Fig. 8 are the output waveforms for the add operation of 1111 + 0001. In this figure, both of the 4-bit inputs (an addend and an augend) were initially set at 0000, then at time = 0, one input was turned to 1111 and the other to 0001. The operation speed is limited by the carry delay, and the mean add time for the operation of 1111 + 0001 was 1.2 ns.

3.2 Parity Generator

Parity generation is the function of detecting whether the number of ones ("1"s) in an input word is odd or even. Figure 9 illustrates a sample circuit for odd parity generation of 9-bit inputs (an 8-bit word plus a parity bit). It was confirmed by simulation that the circuit produces correct operation for all input combinations. Part of the simulated result is illustrated in Fig. 10 with the waveforms for the nine input bits and the output. In this figure, the nine input bits are applied in sequence as (010111100, 010111101, 010111110,

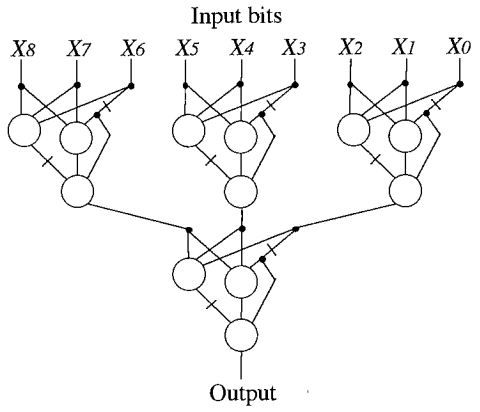


Fig. 9 Circuit for an odd parity generator of 9-bit inputs. The symbol denotes a complementary function gate. An inverter is represented by a segment on a connection branch.

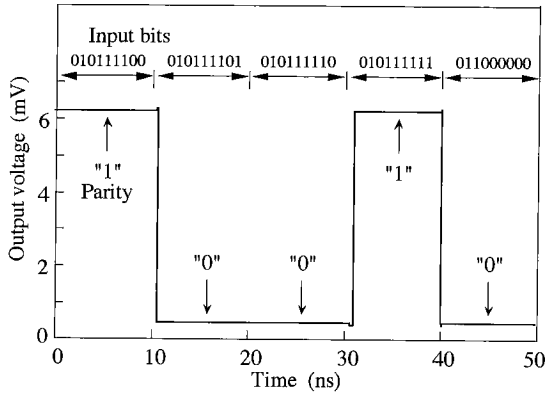


Fig. 10 Operation of the parity generator (simulation). Plotted is the output waveform for the nine inputs applied in sequence as (010111100, 010111101, 010111110, 010111111, 011000000).

010111111, 011000000), and the corresponding output is produced as (1, 0, 0, 1, 0). It can be seen that the circuit produces successfully an output of 1 for an odd number of ones in the inputs. The mean operation time in this circuit was 0.8 ns.

4. Operation Error Caused by Thermal Agitation

In designing single-electron circuits, of special concern are operation errors caused by thermal agitation. The single electron circuit is designed so that a specific tunneling will occur for a given input signal to produce the correct output. But, when a circuit is operated at *non-zero* temperatures, unexpected or unnecessary tunnelings can be induced by thermal agitation to produce operation errors in the circuit.

To characterize thermal error in a single-electron circuit, we here define two indices, the *number of errors* and the *time ratio of errors*. The meanings of the indices are as follows. Consider a single-electron logic circuit with its inputs fixed at a given set of 1-0 values. At zero temperature (0 K), the circuit will produce the

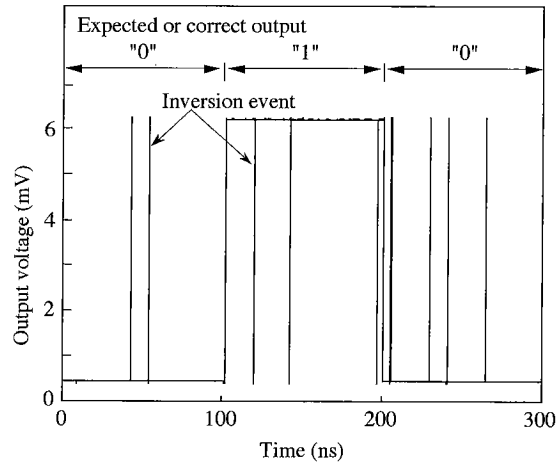


Fig. 11 Simulation result for the MSB output of the adder as a function of time at a temperature of 0.5 K. The expected sequence of the output is (0, 1, 0). Thermal noise is conspicuous.

stable output (1 or 0) that corresponds to the given inputs. At non-zero temperatures, however, unnecessary tunnelings are induced irregularly, and consequently the output of the circuit will produce chattering; that is, at each event of the unnecessary tunnelings, the output will show a temporary inversion in its binary value (therefore a temporary error in logic output). We call this event an *inversion event*. To estimate the degree of operation errors, we observe the circuit output for a fixed period of time to count the inversion events in the output. The *number of errors* means the number of times of inversion events, and the *time ratio of errors* means the ratio of *a total of the duration of each inversion event* to the observation period.

The frequency of thermal errors can be estimated analytically for simple circuits composed of a few tunnel junctions. But a circuit of greater complexity is difficult to calculate on paper, so computer simulation is needed. In the following, we will illustrate the result of error simulation for the 4-bit adder circuit discussed in Sect. 3.1.

We first simulated the time dependence of the adder output for various input combinations and temperatures. A portion of the results is illustrated in Fig. 11. In this instance, one input for the adder was fixed at 0101 and the other input was applied in sequence as (0010, 1010, 1110), assuming a temperature of 0.4 K. Illustrated in the figure is the waveform for the most-significant bit (MSB) of the sum output. The expected sequence of the MSB output is (0, 1, 0), but owing to thermal noise, the output is unsettled in practice by frequent occurrence of inversion events induced by unnecessary tunnelings. (The other three bits of the sum output and the bit of the carry output show similar behavior to the MSB in respect to thermal noise, so they are not illustrated here.)

To see the degree of operation error, we calculated

the two indices (the number of errors and the time ratio of errors) for the simulation data of the MSB output. Properly speaking, operation error must be estimated in consideration of overall adder operation for all possible input combinations. For simplicity, however, we will consider here the thermal error only for the MSB output for a given instance of adder inputs.

We simulated the time dependence of the MSB output for various temperatures, assuming adder inputs of 0101 and 0010. The expectation value of the MSB is 0, but in practice the MSB output is disturbed frequently by thermal noise. We calculated the number of errors and the time ratio of errors by observing the MSB output for 10 ns and measuring the inversion events in the output. We call this procedure *a trial calculation*. For a given value of temperature, we repeated 100 trial calculations for different sets of random numbers and then calculated the mean value for the trials. In Fig. 12, the calculated mean value is represented by an open circle (for the number of errors) and a solid circle (for the time ratio of errors), where a vertical segment transpiercing the circle indicates the maximum and the minimum value in the 100 trials. It can be seen that operation error increases rapidly with increase in temperature. The frequency of thermal noise depends on circuit parameters and the length of time of the observation period, so we cannot jump to conclusions. Still, we here make a rough estimation from the figure. First, if we permit no inversion event, the adder must be operated at 0.2 K or less. Second, if we judge the output logic value from the average of outputs for a fixed period of time, then several inversion events are allowable to the extent that the time ratio of errors is sufficiently smaller than 0.5. Therefore the adder can be used at temperatures up to 1 K.

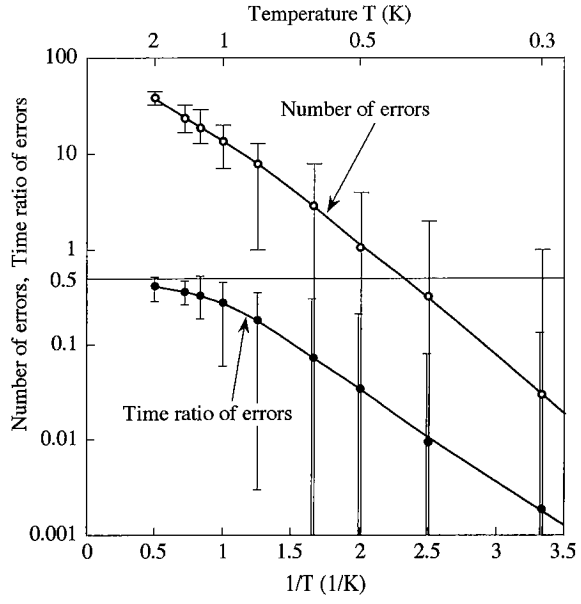


Fig. 12 The number of errors and the time ratio of errors in the MSB output of the adder. Illustrated are the results obtained by observing the MSB output for 10 ns. For details, see the text.

5. Conclusion

As a promising device for constructing integrated circuits based on single-carrier electronics, we proposed the single-electron majority logic circuit. The majority logic method is a way of implementing digital operations in a manner different from that of Boolean logic, and can implement a given logic function with a smaller number of devices.

We first proposed a gate circuit for implementing the unit function required for majority logic operation. It consist of a capacitor array for input summation and a single-electron inverter for threshold operation. The circuit accepts an odd number of input voltages and produces the corresponding output voltage on the basis of the principle of majority decision; it produces an output of logic "1" ("0") if the majority of the inputs is 1 ("0"). The issues were also discussed that require attention in determining the parameters of the gate circuit for achieving correct operation.

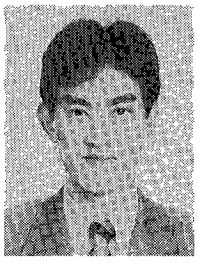
To show the possibility of constructing various subsystems, we then designed a 4-bit adder and a 9-bit parity generator by combining the proposed gate circuits, then showed by computer simulation that the designed subsystems produce the correct logic operations. The operation error induced by thermal agitation was also estimated, taking the adder as an example. It was shown that correct operation can be expected at temperatures of up to 1 K. We will be able to construct large logic systems by use of the proposed single-electron majority logic circuit.

(This work was supported by a Grant-in-Aid for Scientific Research on Priority Areas "Single-electron devices" from the Ministry of Education, Science, Sports and Culture, and by the Proposal-Based New Industry Creative Type Technology R&D Promotion Program from the New Energy and Industrial Technology Development Organization (NEDO).)

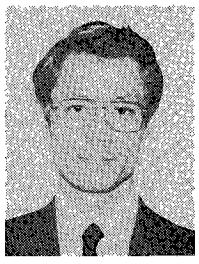
References

[1] S. Amarel, G. Cooke, and R.O. Winder, "Majority gate network," IEEE Trans. Electronic Computers, vol.13, no.1, pp.4–13, 1964.
[2] A.R. Meo, "Majority gate networks," IEEE Trans. Electronic Computers, vol.15, no.4, pp.606–618, 1966.
[3] J.R. Tucker, "Complementary Digital Logic Based on the "Coulomb Blockade"," J. Appl. Phys., vol.72, no.9, 1992.
[4] M. Akazawa and Y. Amemiya, "Eliciting the potential functions of single-electron circuits," IEICE Trans. Electron., vol.E80-C, pp.849–858, 1997.
[5] N. Kuwamura, K. Taniguchi, and C. Hamaguchi, "Simulation of single-electron logic circuits," IEICE Trans., vol.J77-C-II, no.5, pp.221–228, 1994.

Hiroki Iwamura was born in Sapporo on Augest 7, 1973. He received the B.E. degree from Hokkaido University in 1996. He is currently working toward the M.E. degree in electrical engineering. He is carrying out researches on single-electron digital circuits based on majority logic, threshold logic, and multi valued logic.



Masamichi Akazawa was born in Hokkaido, Japan, on March 22, 1966. He received the B.E. and Dr. Eng. degrees in electrical engineering from Hokkaido University in 1988 and 1994, respectively. He joined the Faculty of Engineering, Hokkaido University as a Research Associate, and is currently an Associate Professor in the Faculty of Electrical Engineering. His current research is in the field of new functional devices/LSIs, single-electron phenomena, quantum structures, neural network architectures, and intelligent sensors.



Yoshihito Amemiya was born in Tokyo, Japan, on March 5, 1948. He received the B.E., M.E., and Dr. Eng. degrees from the Tokyo Institute of Technology in 1970, 1972, and 1975, respectively. From 1975 to 1993, he was a Member of the Research Staff at NTT LSI Laboratories, Atsugi, Japan. Since 1993 he has been a professor in the Faculty of Electrical Engineering at Hokkaido University. His research is in the field of semiconductor devices and circuits, digital- and analog-processing elements utilizing quantum phenomena and single-electron effects, neural network devices and circuits, and intelligent material systems and structures.