# SINGLE-FLUX-QUANTUM LOGIC DEVICES BASED ON THE BINARY DECISION DIAGRAM

Noboru Asahi, Takashi Yamada, Masamichi Akazawa, and Yoshihito Amemiya

Faculty of Engineering Hokkaido University, Kita 13, Nishi 8, Sapporo 060-8628, Japan

**Abstract:** This paper proposes single-flux-quantum logic devices based on the concept of the binary decision diagram (BDD). The unit device consists of several Josephson junctions and operates as a two-way switch for single-flux-quantum transport. Any combinatorial logic can be implemented by connecting identical unit devices into a cascade to build the tree of a BDD graph. The construction of an 8-bit adder system is presented as an example.

**Key words:** *binary decision diagram, BDD, single flux quantum, SFQ, logic circuit*

## Introduction

One of the promising areas of research in superconductor electronics is the development of circuit technologies that can be used in constructing large digital processing systems. To promote such technologies, we propose a method of *constructing single-flux-quantum circuits on the basis of the binary decision diagram (BDD)*. The BDD is a graphical method for representing digital functions and can provide a concise expression for most logic functions encountered in LSI design applications [1][2]. The authors suggested in previous articles that the concept of the BDD can be utilized for constructing various quantum-device circuits such as single-electron circuits [3]-[5]. In the present paper, we will describe a method of constructing single-flux-quantum BDD circuits by using an 8-bit adder system as an example. By using single-flux-quantum BDD circuits, we will be able to construct various logic systems that are capable of high-speed operation [6].

## Binary decision diagram (BDD)

A BDD is a directed graph composed of many nodes and two terminals, with each node labeled with a variable (Fig. 1). In determining the value of the function for a set of input variables, we enter at the root of the graph and proceed downward to a terminal. At each node, we follow the branch corresponding to the value of the variables. The logic value of the function is determined by which terminal we reach. In the BDD illustrated in Fig. 1(b), a set of functions is represented efficiently by a single graph with multiple roots (one for each function). This type of BDD is called a shared BDD. In a BDD, for a given set of the variables, there is only one path from the root to either terminal of 1 or 0. Therefore, we can also determine the function value by tracing the path upward from a terminal to the root of the graph. If there is a path from the 1-terminal to the root, the function value is 1; otherwise, the function value is 0.

## Single-flux-quantum BDD device

A BDD consists of many nodes, and the function of each node is two-way switching controlled by an input variable. To implement this function using single-flux-quantum technology, we propose the device element illustrated in Fig. 2 (we will call this element a *BDD device*). The BDD device consists of two control gates (*J0*, *J1*) and three buffer circuits. The gates are controlled by a binary input signal $Xi$ (and its complement $\overline{Xi}$). The operation required for the control gates is as follows: if $Xi = 1$ ($\overline{Xi} = 0$), the gate *J1* is in a superconducting state (a state of high critical current) and *J0* is in a resistive state (a state of low critical current), and if $Xi = 0$ ($\overline{Xi} = 1$), the gate *J1* is in a resistive state and *J0* is in a

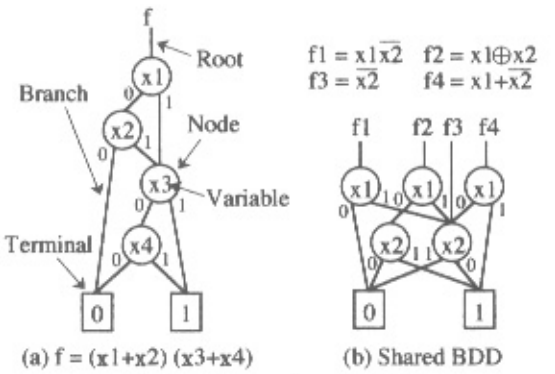


**Fig. 1.** Examples of BDD.

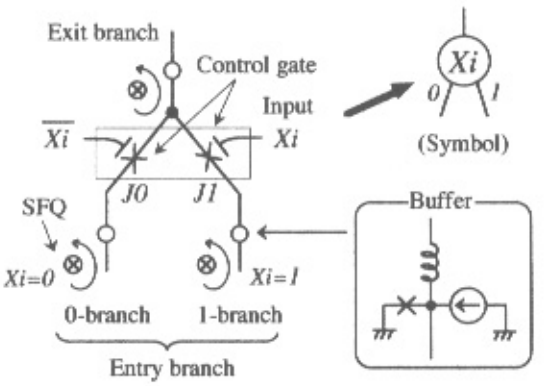


**Fig. 2.** The BDD device consisting of a single-flux quantum circuit.

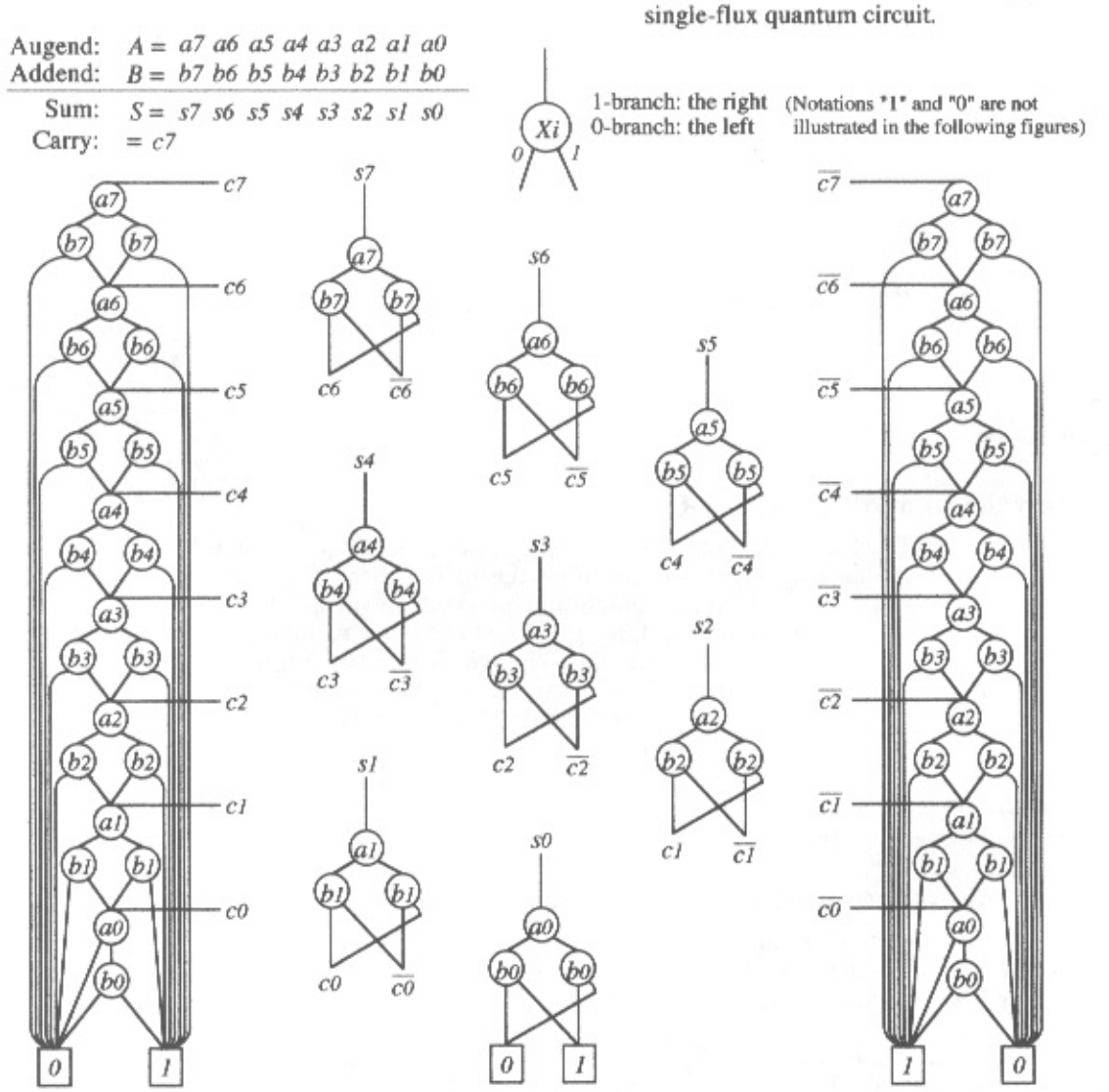


**Fig. 3.** A shared-BDD representation for an 8-bit adder: s0 through s7 denote the sum output bits; c0 through c7 denote the carry output bits; and $\overline{c0}$ through $\overline{c7}$ denote the complements for c0 through c7.

superconducting state. Several methods can be employed for constructing the control gate. The candidate methods are: (i) using a simple Josephson junction and controlling its critical current by using an input current, and (ii) using a Josephson MOS transistor and controlling its critical current by using an input voltage.

The BDD device operates as a two-way switch that accepts a single-flux-quantum pulse (SFQ pulse) from either entry branch and transports the pulse to the exit branch. For an input of $Xi = 1$, the device transports a SFQ pulse if and only if the pulse is injected through the 1-branch (a SFQ pulse through the 0-branch is damped down by the control gate *J0*). And if $Xi = 0$, the device transports a SFQ pulse only from the 0-branch to the exit branch.

## Construction of the BDD circuit

Any combinatorial logic circuit can be constructed by connecting the BDD devices into a cascade to build the tree of a BDD graph. We here present a sample design for an 8-bit

**Table 1.** Design parameters

| | | | |
|---|---|---|---|
| Bias current | 0.1 mA | Junction conductance | 0.2 S |
| Inductance | 8 pH | (Junction conductance for the entry buffers in the BDD device is 0.4 S.) | |
| Output resistance | 3 Ω | | |
| Junction capacitance | 0.1 pF | | |
| Critical current | 0.13 mA | | |

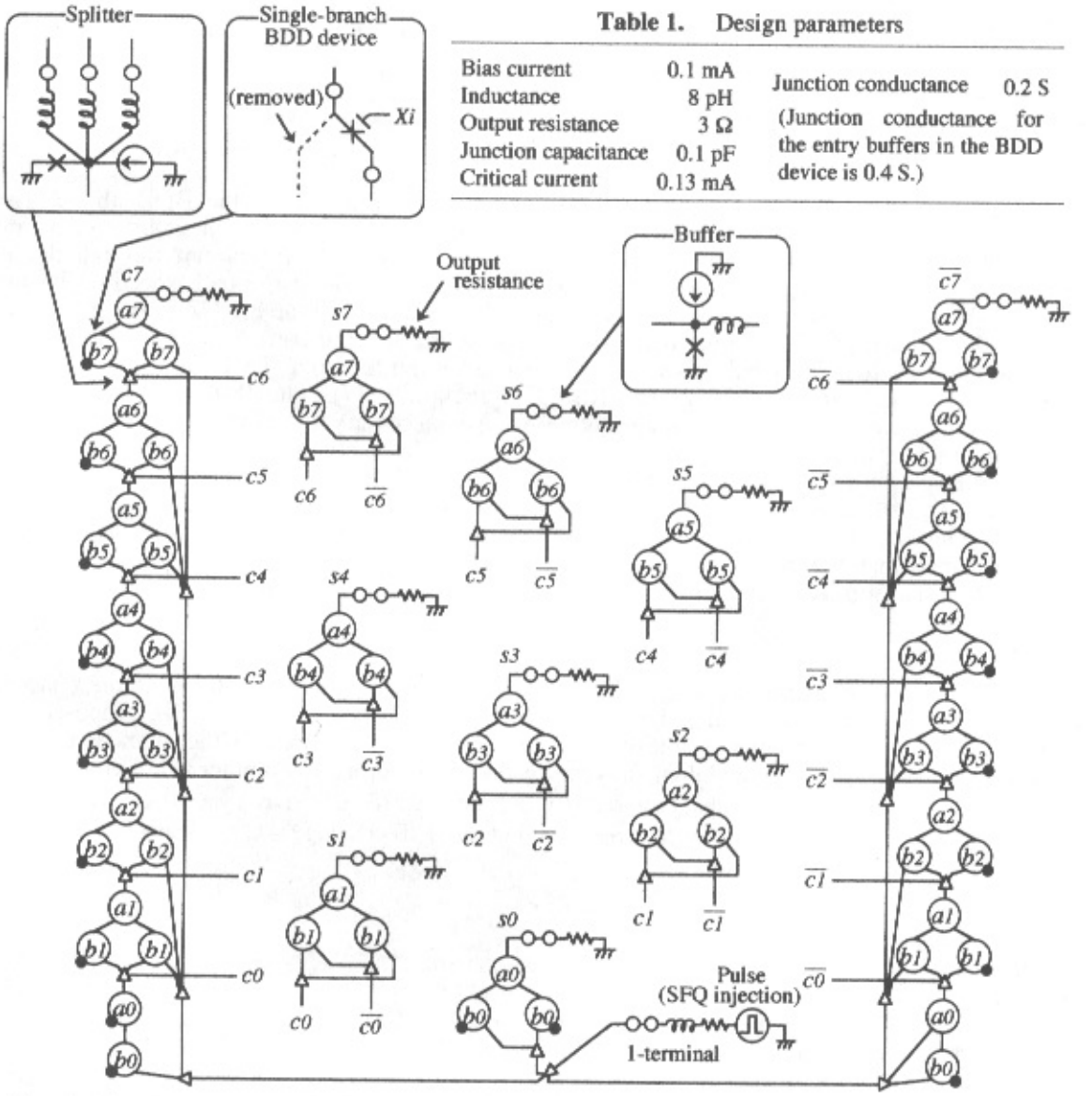


**Fig. 4.** The single-flux-quantum BDD circuit for an 8-bit adder. This circuit is obtained by replacing the nodes in the BDD representation with the BDD devices.

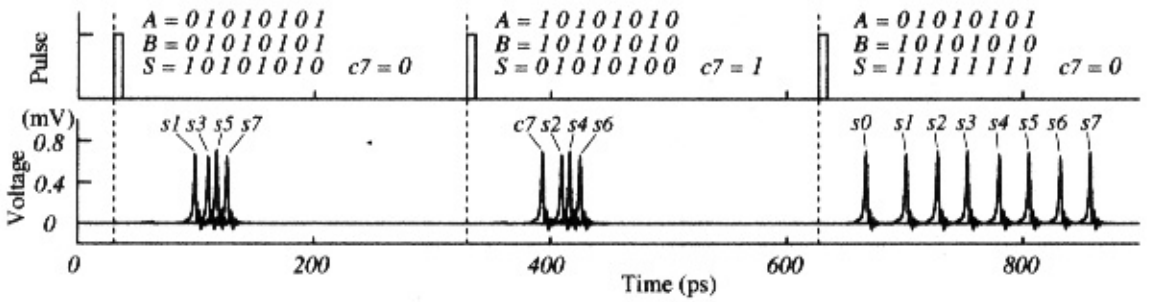


**Fig. 5.** Simulation results for the operation of the adder circuit in Fig. 4. Plotted are the waveforms for the output voltage (SFQ pulse output) on each root.

adder. It accepts two 8-bit inputs (an addend and an augend) and produces the corresponding 8-bit sum output and 1-bit carry output. The operation of the adder is represented by the BDD illustrated in Fig. 3 (the ripple-carry type is shown). For this representation, we used a shared BDD with nine roots (one for each output bit). The BDD consists of sum-bit subgraphs (*s7*, ..., *s0*) and carry-bit subgraphs (c7, ..., *c0*, $\overline{c7}$, ..., $\overline{c0}$). The signals $\overline{c7}$ through $\overline{c0}$ are the complements for the carry output *c7* and the internal carry signals *c6* through *c0*.

We designed a single-flux-quantum logic circuit that implements the BDD above. The configuration of the circuit is illustrated in Fig. 4. In add operation, the input bit signals are applied to the BDD devices, then an SFQ pulse is injected into the circuit through the 1-terminal. The SFQ pulse is transported upward to each root along the path specified by the inputs, being reproduced by the splitters. The value of each output bit (*s7* through *s0*, and c7) is determined by observing whether a SFQ pulse reaches the corresponding root; i.e., the output bit is 1 if a SFQ pulse reaches the root. The splitters and the buffers are the same circuit elements that are used in the RSFQ logic family [7]. The 0-terminal and related branch connections in the BDD representation are unnecessary for circuit construction. The single-branch BDD device in Fig. 4 is a BDD device that has a single entry branch.

We have confirmed by computer simulation that the designed adder performs correct add operation for all possible input combinations. Part of the simulation results is illustrated in Fig. 5. The circuit parameters used are given in Table 1. For the control gate junctions, it was assumed that the critical current of the junction is unchanged for an input of 1 and is reduced to one fifth for an input of 0. The circuit operated in 200-ps add time.

## Conclusion

We proposed constructing single-flux-quantum logic circuits on the basis of the concept of the binary decision diagram (BDD). To give practical form to this idea, we proposed the device element that implements the unit BDD function. By combining the element devices we designed an 8-bit adder, and confirmed correct add operation by computer simulation.

(This work was supported 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. B. Akers, *IEEE Trans. Comput.*, vol. C-27, No. 6, pp. 509-516, 1978.
2. R. E. Bryant, *IEEE Trans. Comput.*, Vol. C-35, No. 8, pp. 677-691, 1986.
3. N. Asahi, M. Akazawa, and Y. Amemiya, *IEEE Trans. ED.*, Vol. 42, No. 11, pp. 1999-2003, 1995.
4. N. Asahi, M. Akazawa, and Y. Amemiya, *IEEE Trans. ED.*, vol. 44, No.7, pp. 1109-1116, 1997.
5. N. Asahi, M. Akazawa, and Y. Amemiya, *IEICE Trans. Electronics*, E81-C, pp. 49-56, 1998.
6. N. Asahi, M. Akazawa, and Y. Amemiya, *Technical Report of IEICE*, SCE 98-20, pp. 49-56, 1998.
7. K. K. Likharev and V. K. Semenov, *IEEE Trans. Appl. Supercond.*, vol. 1, No. 1, pp. 3 -26, 1991.