Advanced Design Approaches for SFQ Logic Circuits Based on the Binary Decision Diagram

T. Nishigai, M. Ito, N. Yoshikawa, K. Obata, K. Takagai, N. Takagai, A. Fujimaki, H. Terai, and S. Yorozu

Abstract—We have been investigating a design methodology of SFQ logic circuits based on the binary decision diagram (BDD). In the previously proposed BDD SFQ logic circuits, we have used one-to-two binary switches as a node cell in a BDD tree. In this study we will propose a new implementation method of SFQ BDD circuits, in which two nodes are implemented by using a 2-input 2-output switch gate. By employing the new approach, we have designed and implemented a one-bit full adder using the NEC 2.5 kA/cm² Nb standard process and the CONNECT cell library. The maximum operating frequency of the full adder was found to be 40 GHz by circuit simulations and 32.8 GHz by on-chip high-speed tests.

Index Terms—Adder, asynchronous circuit, BDD, binary decision diagram, dual rail, SFQ logic circuit, superconducting circuit.

I. INTRODUCTION

SINGLE FLUX QUANTUM (SFQ) circuits have been developed widely because of their potentially high performance with a high clock frequency and low-power consumption [1]. Typical SFQ circuits, however, need distribution of clock signals to each logic gate, and the timing design of large-scale SFQ circuits is expected to become more difficult due to relatively large clock skew at high clock rates. One solution to overcome the timing issue is the introduction of an asynchronous design approach that eliminates a global clock for individual logic gates [2]. We have proposed an asynchronous design approach based on the binary decision diagram (BDD) for SFQ circuits and shown its validity previously [3]. The BDD SFQ logic style is now used in the design of circuit blocks of recent large-scale SFQ circuits. For example, the controller of the SFQ microprocessor, CORE1 α [4] uses this design style.

In the BDD SFQ circuits proposed earlier, we have used a one-to-two binary switch gate as a node cell in a BDD tree. Though the BDD design approach helps to reduce the circuit design complexity and the circuit delay, one problem we faced sometimes is its limited throughput, which arises from the large

Fig. 1. An example of a binary decision diagram (BDD), which represents a one-bit full adder.

setup/hold times and the unequal delays at the complementary outputs of the node cell.

In this study, we will propose two new methods to implement the BDD SFQ circuits. The first approach uses the 2×2 -Join, which is one of primitive gates in the delay insensitive SFQ circuit [2], to represent a pair of nodes in a BDD tree. Second approach uses a D flip-flop with two clock inputs and complementary outputs, which is called d2ff in our cell library. We will compare the new approaches with the previous one in terms of operational margins, throughput and latency using the circuit simulations. We will also design and implement a one-bit full adder by employing the d2ff as a BDD node cell to demonstrate its high-speed operation by an on-chip high-speed test.

II. BDD SFQ LOGIC CIRCUITS

The BDD is a way to represent a logical function by a directed graph, which is composed of nodes having one input and two outputs [3]. The node switches an input signal into one of the two outputs depending on its internal state. An example of the BDD tree is shown in Fig. 1, which represents a one-bit full adder. In the figure, an input signal applied to the top of the graph propagates through the nodes into output terminals denoted by “0” or “1”, which correspond to an output value of the logical function.

In the BDD SFQ circuits, an SFQ pulse propagates through the BDD tree without loss of its amplitude. In the previously proposed BDD SFQ circuits, we have implemented BDD nodes by using one-to-two binary switch gates [3]. More diagram of these gates are shown in Fig. 2. The Bina is a resettable, destructive read-out D flip-flop with complementary outputs [3]. The ndroc is resettable, nondestructive read-out D flip-flop with complementary outputs [5]. These gates are available in the CONNECT cell library [6] as a standard cell.

Manuscript received October 5, 2004. This work was supported by the NEDO through ISTECS as a Collaborative Research and Superconductors Network Device Project.

T. Nishigai, M. Ito, and N. Yoshikawa are with the Department of Electrical and Computer Engineering, Yokohama National University, Yokohama 240-8501, Japan (e-mail: yoshi@yoshilab.dnj.ynu.ac.jp).

K. Obata, K. Takagai, and N. Takagai are with the Department of Information Engineering, Nagoya University, Nagoya 464-8603, Japan.

A. Fujimaki is with the Department of Quantum Engineering, Nagoya University, Nagoya 464-8603, Japan.

H. Terai is with the National Institute of Communications Technology, Kobe 651-2492, Japan.

S. Yorozu is with ISTECS-SRL, Tsukuba 305-8501, Japan.

Digital Object Identifier 10.1109/TASC.2005.849855

Fig. 2. One-to-two binary switch gates for BDD SFQ circuits. (a) The Bina. (b) The ndroc.

Fig. 3. (a) A partial graph of a generalized BDD tree. (b) Its implementation using one-to-two binary switch gates.

A partial graph of a generalized BDD tree and its implementation by the one-to-two binary switch are shown in Fig. 3(a) and (b), respectively. In Fig. 3(b), “set1” or “set0” is provided to two nodes, node 1 and node 2, to set the internal state of the nodes. Then an SFQ pulse is applied from the upper level to node 1 or node 2 as a clock signal. Depending on its internal state, an SFQ pulse is generated to one of the four terminals connected to the lower level. Though the implementation of the BDD by the one-to-two switch gates is simple, it turns out that these gates have disadvantages of the large setup/hold times and the inequality of the delays in the complementary outputs, which limit the throughput of the circuits.

III. NEW DESIGN APPROACHES FOR BDD SFQ CIRCUITS

New approaches proposed here use two-by-two switches to implement a pair of nodes in a BDD tree. Fig. 4 shows two logic gates, the 2×2 -Join [2] and the d2ff, which will be used as two-by-two switch gates for new BDD SFQ circuits.

The 2×2 -Join is a two-by-two switch with four outputs, “d00”, “d01”, “d10” and “d11”. An output pulse is obtained from one of the four outputs depending on the combination of its two complementary inputs (“At”, “Af”) and (“Bt”, “Bf”). The

Fig. 4. Two-by-two switch gates for the BDD SFQ circuits. (a) The 2×2 -Join and (b) the d2ff. A dotted line shows a storage loop.

outputs are given by $d00 = Af \wedge Bf$, $d01 = Af \wedge Bt$, $d10 = At \wedge Bf$ and $d11 = At \wedge Bt$. The details of the circuit schematic and its operation are described in [2].

The d2ff is a destructive read-out D flip-flop with two clock inputs, “clk1” and “clk2”. As shown in the circuit schematic of Fig. 4(b), $J_1-L_2-L_3-J_6-L_{10}-L_6-J_4$ and $J_1-L_2-L_3-J_7-L_{11}-L_7-J_5$ compose a storage loop of the flux. In the initial state, an SFQ pulse applied to “din” sets the internal state of the gate. Then if an SFQ pulse is applied to one of the two clock terminals, an output signal is obtained from “d1” or “d2”. The outputs are represented by $d1 = din \wedge clk1$ and $d2 = din \wedge clk2$, where “din” has to be applied before the clock. Two d2ff gates compose a two-by-two switch, sharing clk1 and clk2 in each gate. In this case, outputs (d11, d12) of “d2ff 1” and outputs (d21, d22) of “d2ff 2” are represented by $d11 = din1 \wedge clk1$, $d12 = din1 \wedge clk2$, $d21 = din2 \wedge clk1$, and $d22 = din2 \wedge clk2$, which are similar to the function of the 2×2 -Join. Both the 2×2 -Join and the d2ff are available in the CONNECT cell library.

The partial graph of a generalized BDD given in Fig. 3(a) is implemented by using the 2×2 -Join and the d2ff as shown in Fig. 5. In Fig. 5(a), node 1 and node 2 are implemented by one 2×2 -Join. An input “set1” or “set0” applied to “At” or “Af”, respectively, sets the internal state of the nodes. Then if an SFQ pulse from the upper level is applied to “Bt” or “Bf”, an SFQ pulse is generated from one of the four output terminals “n1”, “n2”, “n3” and “n4”.

Fig. 5(b) shows another implementation of two-by-two switches using the d2ff. In this case an SFQ input, “set1” or “set0”, is applied to “din” of one of two d2ff gates to set its internal state. Then if an SFQ pulse is given from upper level to “clk1” or “clk2”, an SFQ pulse is generated from one of the four terminals “n1”, “n2”, “n3” and “n4”.

Fig. 5. Implementations of a partial graph of a generalized BDD tree by using (a) the 2×2 -Join and (b) the d2ff.

TABLE I
DC BIAS MARGINS AND CRITICAL MARGINS OF THE LOGIC GATES

Logic cell	Number of JJ	DC bias margin [%]	Critical margin [%]
d2ff	24	-35.9 ~ +45.3	+/- 26.6
2x2-Join	20	-29.7 ~ +23.4	+/- 18.8
ndroc	66	-29.7 ~ +29.7	+/- 23.4
Bina	38	-29.7 ~ +35.9	+/- 18.8

IV. COMPARISON OF SWITCH GATES

In this section, we will compare basic properties of the switch gates for BDD SFQ circuits in terms of operational margins, throughput and latency using circuit simulations. All logic gates were optimized by the critical margin method by using the SFQ circuit optimizer, SCOPE and were included in the CONNECT cell library as standard cells [6].

A. Operational Margins

DC bias margins and critical margins are good measures to consider the robustness of the SFQ logic gates. The DC bias margin is defined as the range of the global DC bias current for the correct circuit operation, whereas the critical margin is the narrowest range of all the circuit parameters for the correct circuit operation. These values for each switch gate are listed in Table I, where we assumed the input data rate was 5 Gbps. Table I also shows the total junction number to implement a pair of BDD nodes. One can see that the d2ff has the largest DC bias margin and critical margin. The 2×2 -Join has the smallest junction counts to make BDD SFQ circuits, though its DC bias margin is rather small.

B. Throughput

The dependence of the setup and hold times on the DC bias current was calculated to evaluate the throughput of the switch gates. The definition of the setup and hold times is illustrated in Fig. 6(a). The calculation results are shown in Fig. 6(b), where the region between the setup time and the hold time corresponds to the time window in which the data can be applied without

Fig. 6. (a) A definition of the setup and hold time. (b) The dependence of the setup and hold time of the switch gates on the DC bias current.

malfunction. In the figure the clock period is assumed to be 50 ps. This time window should be larger to achieve higher throughput. One can see that the d2ff has the smallest setup/hold times with almost no DC bias dependence and the large time window.

C. Propagation Delay

We have calculated the dependence of the propagation delay on the DC bias current (Fig. 7(a)) to consider the latency of the switch gates. The propagation delay should be smaller to achieve smaller latency. We have also calculated the difference of the delays of the switch gate at the complementary outputs (Fig. 7(b)). The smaller DC bias current dependence and the equality of the delays at the complementary outputs are necessary to increase the throughput and to reduce the timing design complexity. One can find that the d2ff has good properties both in the propagation delay and in the equality of the delays.

V. ON-CHIP HIGH-SPEED TEST

The comparison of basic properties of the switch gates in the previous section indicates that the d2ff is the best choice for the implementation of the BDD SFQ circuits. In order to show the validity of the d2ff as a BDD node, we have designed a one-bit full adder based on the BDD tree shown in Fig. 1 and implemented an on-chip high-speed test system [7] using NEC 2.5 kA/cm² Nb standard process and the CONNECT cell library.

The system was composed of three 4-bit DDST shift registers [7] to load three input data, two 4-bit DDST shift registers to store two calculation results from the BDD adder, and a 4-bit clock generator (CG) to provide a high-speed clock. The system

Fig. 7. The bias current dependence of (a) the propagation delay and (b) the difference of the delays between complementary outputs of the switch gates.

contained 1788 Josephson junctions and its size was $2.72 \text{ mm} \times 1.48 \text{ mm}$. We have examined the DC bias margins of the BDD adder at various frequencies by changing the bias current for the CG separately.

The frequency dependence of the DC bias margins of the BDD full adder using the d2ff is shown in Fig. 8. We have estimated the frequency of the CG by the circuit simulation. A good agreement has been obtained between the measured and calculated frequencies of the CG in the previous tests [8]. In Fig. 8 the calculation results of circuit simulations and logic simulations are also shown for comparison. It can be seen that the test results agree well with the simulation results. It indicates that the accuracy of the circuit and logic simulation is quite good. The maximum operating frequency was found to be 40 GHz from the circuit simulations, and 32.8 GHz from the on-chip high-speed test.

Fig. 8. The dependence of the DC bias margins of the BDD full adder using the d2ff on the data input frequency obtained by high-speed tests and simulations.

VI. CONCLUSION

We have proposed a new implementation method of BDD SFQ circuits, which uses a two-by-two switch as a node cell in a BDD tree. We have compared the basic properties of the logic gates for the BDD SFQ circuits in terms of operational margins, throughput and latency, and found that the new approach based on the d2ff is the best choice. We have also designed and implemented a one-bit BDD full adder using the d2ff gates and performed an on-chip high-speed test. Its frequency dependence of the DC bias margins agrees well with the simulation results. The maximum operating frequency of the BDD adder was estimated to be 32.8 GHz from the on-chip high-speed test.

REFERENCES

- [1] K. K. Likharev and V. K. Semenov, "RSFQ logic/memory family: A new Josephson-junction technology for sub-terahertz-clock-frequency digital systems," *IEEE Trans. Appl. Supercond.*, vol. 1, pp. 3–28, Mar. 1993.
- [2] P. Patra, S. Polonsky, and D. S. Fussell, "Delay insensitive logic for RSFQ superconductor technology," in *Proc. Int. Symp. Adv. Res. Asynchronous Circuits Syst.*, Apr. 1997, pp. 42–53.
- [3] N. Yoshikawa and J. Koshiyama, "Top-down RSFQ logic design based on a binary decision diagram," *IEEE Trans. Appl. Supercond.*, vol. 11, pp. 1098–1101, Mar. 2001.
- [4] M. Tanaka, F. Matsuzaki, T. Kondo, N. Nakajima, Y. Yamanashi, A. Fujimaki, H. Hayakawa, N. Yoshikawa, H. Terai, and S. Yorozu, "A single-flux-quantum logic prototype microprocessor," in *Tech. Dig. IEEE ISSCC*, Feb. 2004, pp. 298–299.
- [5] K. Fujiwara, H. Hoshina, J. Koshiyama, and N. Yoshikawa, "Design and component test of RSFQ packet decoders for shift register memories," *Physica C*, vol. 378–381, pp. 1475–1480, Oct. 2002.
- [6] S. Yorozu, Y. Kameda, H. Terai, A. Fujimaki, T. Yamada, and S. Tahara, "A single flux quantum standard logic cell library," *Physica C*, vol. 378–381, pp. 1471–1474, Oct. 2002.
- [7] Z. J. Deng, N. Yoshikawa, S. R. Whiteley, and T. Van Duzer, "Self-timing and vector processing in RSFQ digital circuit technology," *IEEE Trans. Appl. Supercond.*, vol. 9, pp. 7–17, Mar. 1999.
- [8] M. Ito, N. Nakajima, K. Fujiwara, N. Yoshikawa, A. Fujimaki, H. Terai, and S. Yorozu, "Design and implementation of SFQ programmable clock generators," *Physica C*, vol. 412–414, pp. 1550–1554, Oct. 2004.