

Title	RSFQ 4-bit bit-slice integer multiplier
Author(s)	Tang, Guang Ming; Takagi, Kazuyoshi; Takagi, Naofumi
Citation	IEICE Transactions on Electronics (2016), E99C(6): 697-702
Issue Date	2016-06-01
URL	http://hdl.handle.net/2433/226242
Right	Copyright © 2016 The Institute of Electronics, Information and Communication Engineers.
Type	Journal Article
Textversion	publisher

RSFQ 4-bit Bit-Slice Integer Multiplier

Guang-Ming TANG^{†a)}, Nonmember, Kazuyoshi TAKAGI^{†b)}, Member, and Naofumi TAKAGI^{†c)}, Senior Member

SUMMARY A rapid single-flux-quantum (RSFQ) 4-bit bit-slice multiplier is proposed. A new systolic-like multiplication algorithm suitable for RSFQ implementation is developed. The multiplier is designed using the cell library for AIST 10-kA/cm² 1.0- μ m fabrication technology (ADP2). Concurrent flow clocking is used to design a fully pipelined RSFQ logic design. A $4n \times 4n$ -bit multiplier consists of $2n + 17$ stages. For verifying the algorithm and the logic design, a physical layout of the 8×8 -bit multiplier has been designed with target operating frequency of 50 GHz and simulated. It consists of 21 stages and 11,488 Josephson junctions. The simulation results show correct operation up to 62.5 GHz.

key words: multiplier, single-flux-quantum (SFQ), microprocessor, superconducting integrated circuits

1. Introduction

Superconducting rapid single-flux-quantum (RSFQ) circuit technology [1] is expected to be a next generation circuit technology which enables ultra-high-speed computation with ultra-low-power consumption [2]. With the progress of RSFQ fabrication process technology, it has become possible to realize an RSFQ LSI including tens of thousands of Josephson junctions (JJs) [3]. The increase of the wiring layers combined with the passive transmission line (PTL) technology [4] further increases the circuit integration density.

An integer multiplier is one of the most important arithmetic circuits for high-speed processing. An RSFQ 4×4 -bit and an 8×8 -bit parallel multiplier have been designed and fabricated [5], [6]. For practical applications, multipliers with longer operand length, e.g., 24, 32, or more, are desired. An RSFQ 24×24 -bit bit-serial multiplier based on the systolic algorithm proposed in [7] has been designed and fabricated as a component of a single-precision floating-point multiplier [8].

In general, an $m \times m$ -bit multiplication requires m^2 bit-multiplications (ANDs) for generating m m -bit partial products and $m - 1$ m -bit additions for summing up them. Therefore, an $m \times m$ -bit parallel multiplier would include m^2 AND gates and $m(m - 1)$ full adders. A logic design of an RSFQ 32×32 -bit parallel multiplier would consist of more than 70 thousands JJs using the cell library [9] for the

AIST ADP2 fabrication process [10], which is difficult to be implemented on a single die. On the other hand, an RSFQ $m \times m$ -bit bit-serial multiplier would have unacceptably high latency for several applications. We think bit-slice architecture can be a solution.

In this paper, we propose an RSFQ 4-bit bit-slice integer multiplier. The proposed multiplier is based on a newly developed systolic-like multiplication algorithm suitable for RSFQ implementation. A $4n \times 4n$ -bit multiplier is mainly composed of n almost identical systolic cells. Its hardware complexity is much lower than that of a parallel multiplier. To the best of our knowledge, this is the first proposal of an RSFQ 4-bit bit-slice multiplier. Although we let the length of a bit-slice be four in this paper, we can design a bit-slice multiplier with any slice length in the same way.

In the proposed 4-bit bit-slice multiplier, each systolic cell generates four 4-bit slices of partial products, accumulates them with three 4-bit slices from the preceding cell through carry save additions, and produces three 4-bit slices to the succeeding cell. In an RSFQ logic design with concurrent flow clocking, each systolic cell is implemented as a 9-stage pipelined circuit. By overlapping the clock cycles for the latter 7 stages with those for the former 7 stages of the succeeding cell, a low latency is achieved.

For verifying the proposed algorithm and the logic design, a physical layout of the 8×8 -bit (i.e., $n = 2$) 4-bit bit-slice multiplier has been designed with target operating frequency of 50 GHz using the cell library for AIST 10-kA/cm² 1.0- μ m fabrication technology (the AIST ADP2).

The remainder of this paper is organized as follows. In the next section, the algorithm and architecture, and RSFQ logic design details of the proposed 4-bit bit-slice multiplier are described. In Sect. 3, a layout and simulation results of the 8×8 -bit multiplier are shown. Finally, Sect. 4 summarizes our findings and concludes the paper.

2. 4-bit Bit-Slice Multiplier

2.1 Algorithm and Architecture

We consider a $4n \times 4n$ -bit multiplication, $Z = X \times Y$, where n is a natural number, $X (= [x_{4n-1}x_{4n-2}x_{4n-3} \dots x_2x_1x_0])$ is a multiplicand, $Y (= [y_{4n-1}y_{4n-2}y_{4n-3} \dots y_2y_1y_0])$ is a multiplier, and $Z (= [z_{8n-1}z_{8n-2}z_{8n-3} \dots z_2z_1z_0])$ is the product.

Each of the $4n$ -bit multiplicand X and multiplier Y is divided into n slices of 4 bits each. The n pairs of operand slices are input one by one starting from the least significant

Manuscript received October 21, 2015.

Manuscript revised January 6, 2016.

[†]The authors are with the Graduate School of Informatics, Kyoto University, Kyoto-shi, 606–8501 Japan.

a) E-mail: g.tang@lab3.kuis.kyoto-u.ac.jp

b) E-mail: ktakagi@i.kyoto-u.ac.jp

c) E-mail: takagi@i.kyoto-u.ac.jp

DOI: 10.1587/transele.E99.C.697

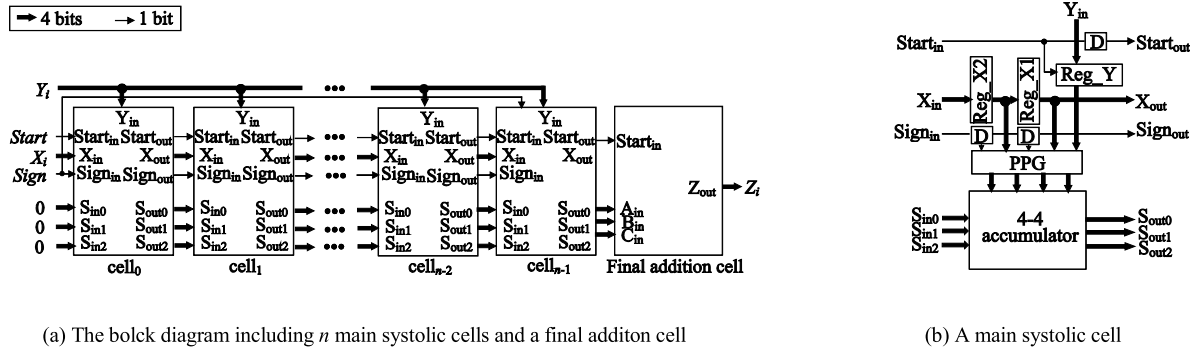


Fig. 1 $4n \times 4n$ -bit 4-bit bit-slice multiplier structural block diagram.

one. The $8n$ -bit product Z is in $2n$ 4-bit slices which are output one by one from the least significant one.

The multiplier performs unsigned or signed integer multiplication. For unsigned integer multiplication, X is multiplied by each bit of Y to generate the partial products, which are added to get Z . Unlike unsigned integer multiplication, signed integer multiplication requires to invert the partial product bits multiplied by the sign bits of X and Y according to the following formula [7]:

$$\begin{aligned}
 Z = & -2^{8n-1} + x_{4n-1}y_{4n-1}2^{8n-2} + \sum_{k=0}^{4n-2} x_{4n-1}y_k 2^{4n+k-1} \\
 & + \sum_{l=0}^{4n-2} y_{4n-1}x_l 2^{4n+l-1} + \sum_{l=0}^{4n-2} \sum_{k=0}^{4n-2} x_l y_k 2^{l+k} + 2^{4n}
 \end{aligned} \quad (1)$$

Therefore, we need to design a control signal to generate the different partial products from unsigned integer multiplication.

The multiplier is based on a newly developed systolic-like algorithm. Figure 1(a) shows a block diagram of a $4n \times 4n$ -bit 4-bit bit-slice multiplier. It is composed of n main cells and a final addition cell. As shown in Fig. 1(b), a main cell consists of a ‘partial product generator (PPG)’ and a ‘4-4 accumulator’, as well as registers, ‘Reg_X1’, ‘Reg_X2’, and ‘Reg_Y’, for keeping two slices of X and a slice of Y , and D flip-flops for keeping signals. The PPG for the most significant cell ($cell_{n-1}$) is slightly different from the others for handling the sign in signed multiplication. The cell also has an additional register for signal $Sign$ which indicates signed multiplication. The final addition cell consists of a 4-bit bit-slice 3-to-2 compressor and a 4-bit bit-slice adder.

A multiplication is carried out through $3n + 1$ logical (systolic) clock cycles. (We use ‘logical cycle’ in order to explain our multiplication algorithm. It is different from the clock cycle of the RSFQ design appearing later.)

The signal $Start$ is fed to $cell_0$ at logical cycle 0 and forwarded to the next cell every logical cycle. (We count the logical cycle from 0.) Therefore, it reaches $cell_i$ at cycle i ($i = 0$ to $n - 1$). The i -th pair of the operand slices, i.e., $X_i (= [x_{4i+3}x_{4i+2}x_{4i+1}x_{4i}])$ and $Y_i (= [y_{4i+3}y_{4i+2}y_{4i+1}y_{4i}])$, is input to

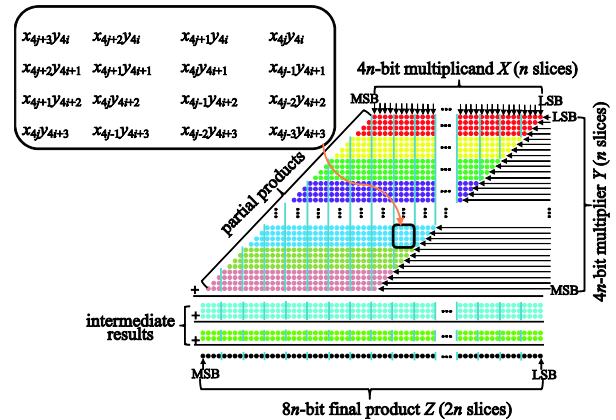


Fig. 2 Dot diagram of $4n \times 4n$ -bit multiplication using 4-bit bit-slice architecture.

the multiplier at cycle i . X_i is fed to $cell_0$ and forwarded to the next cell in every two cycles. Y_i is set to Reg_Y of $cell_i$ by the $Start$ signal and is kept there. The signal for signed multiplication $Sign$, which is 1 for signed multiplication, is fed to the multiplier at logical cycle $n - 1$ with the most significant operand slices, and is immediately split into two. One is fed to $cell_0$ and forwarded to the next cell every two cycles along with X_{n-1} . The other is sent to $cell_{n-1}$ directly, and is set into the additional register by the $Start$ signal. Signal $Sign$ is used for the bit-complementation and generation of the correction terms -2^{8n-1} and 2^{4n} in Eq. (1).

The PPG of $cell_i$ generates four 4-bit slices of partial products at logical cycle $2i$ to $2i + n$. At logical cycle $2i + j$ ($j = 0$ to n), it generates the four 4-bit slices shown in the square in Fig. 2. (Fig. 2 shows a dot diagram of the proposed multiplication. A dot presents a partial product bit or a bit of intermediate results or a final product bit.) Note that at this cycle, Reg_X1 and Reg_X2 hold X_{j-1} and X_j , respectively. The PPG of each main cell generates four partial products (four rows of partial product dots in the same color in Fig. 2).

The 4-4 accumulator of $cell_i$ sums up the four 4-bit partial product slices from the PPG and three 4-bit slices from $cell_{i-1}$ (input from S_{in0} , S_{in1} , and S_{in2}) through carry save additions, and produces three 4-bit slices (output to S_{out0} , S_{out1} , and S_{out2}) at logical cycle i to $2i + n$. Note that at logical cycle i to $2i - 1$, there are no inputs from the PPG.

In the final addition cell, the 4-bit bit-slice 3-to-2 compressor sums up the three 4-bit slices from $cell_{n-1}$ and produces two 4-bit slices, and then, the 4-bit bit-slice adder sums up these two 4-bit slices and produces a 4-bit slice of the final product at logical cycle $n + 1$ to $3n$.

2.2 RSFQ Logic Design

A fully pipelined synchronous RSFQ logic design of the proposed multiplier using concurrent flow clocking is considered. Namely, each pipeline stage consists of a row of RSFQ clocked logic gates. The basic RSFQ logic gates and flip-flops: AND, XOR, NOT, DFF, and NDRO (non-destructive read-out), and wiring elements: JTL (Josephson transmission line), SPL (splitter), CB (confluence buffer), and PTL in the cell library [9] for the AIST ADP2 [10] are used.

Reg_Y is implemented using four NDROs. Reg_X1

and Reg_X2 are implemented using DFFs. PPG is implemented as a row of 16 AND gates. Figure 3 shows that a 4-4 accumulator consists of four 4-bit carry save adders each of which is a row of four full adders (FA's). In the figure, the upper side is the LSB side and the lower side is the MSB side. The three 4-bit slices from the preceding cell are input from S_{in0} , S_{in1} and S_{in2} . The produced three 4-bit slices are output to S_{out0} , S_{out1} and S_{out2} . A full adder is implemented using two AND gates, two XOR gates and a CB. In order to keep the carries from the most significant position of the corresponding slices and add them to the least significant position of the succeeding slices, the DFFs are required.

Figure 4 shows a logic gate level circuit of a 4-4 accumulator with a PPG. As shown in the figure, a main cell consists of 9 stages. Since a full adder is with 2-stage, adjacent cells are aligned in two stages off. Namely, the clock cycles for the latter 7 stages are overlapped with those for the former 7 stages of the succeeding cell.

Figure 5 shows a logic gate level circuit of the final addition cell. The 3-to-2 compressor is nothing but a carry save adder, i.e., a row of four full adders. To reduce the latency of the 4-bit bit-slice adder, a type of parallel prefix (or carry look-ahead) adder called Sklansky adder is used. It consists of 6 pipeline stages. The delay in the feedback loop for the carry signal to the succeeding slice is minimized using the technique developed in [11].

The $4n \times 4n$ -bit multiplier consists of $2n + 17$ stages in total, where two stages for setting Y_0 to $cell_0$, $2n + 7$ for n main cells and 8 for the final addition cell. n pairs of multiplicand and multiplier slices are fed at the first to the n -th clock cycles, and $2n$ slices of the resultant product is output

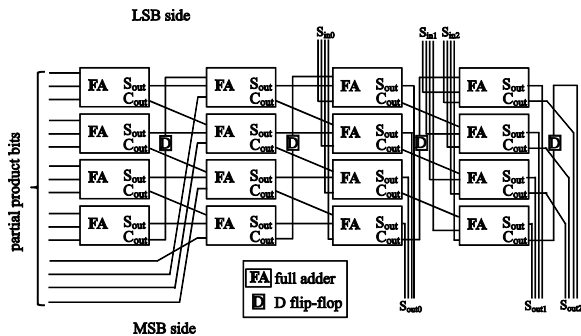


Fig. 3 A block diagram of the 4-4 accumulator.

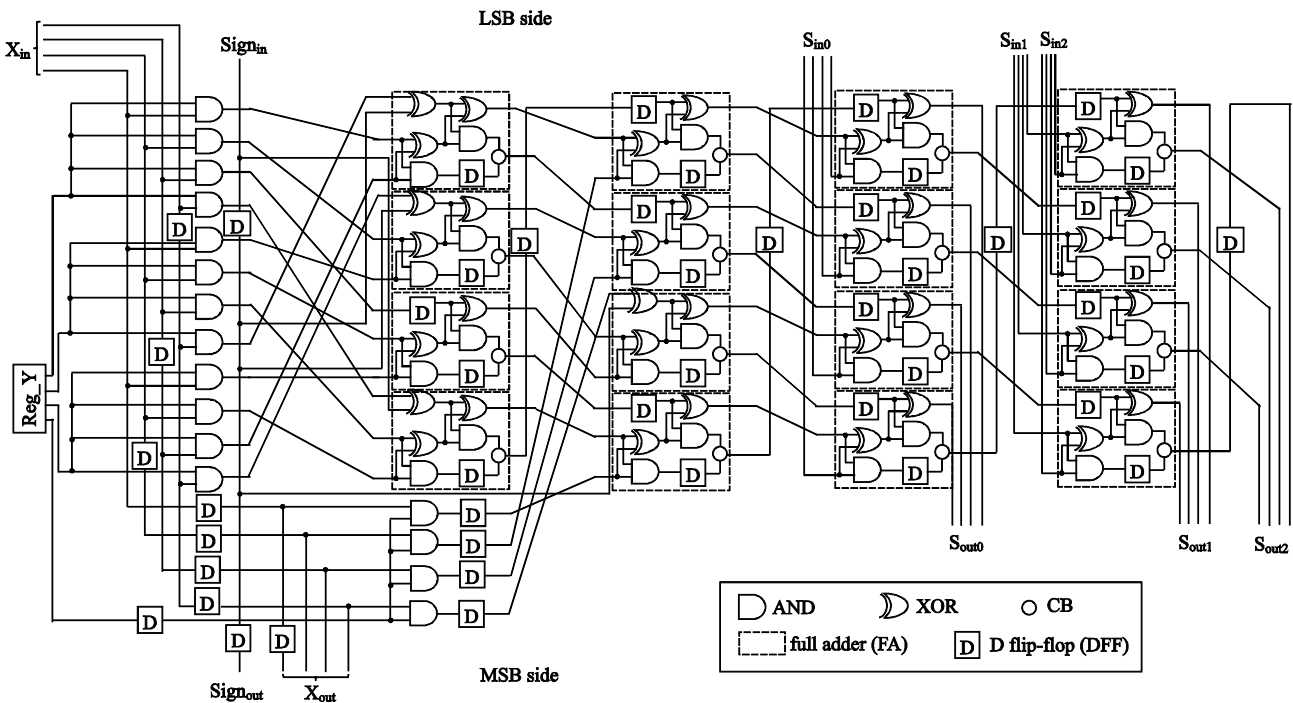


Fig. 4 A logic gate level circuit of a 4-4 accumulator with a PPG.

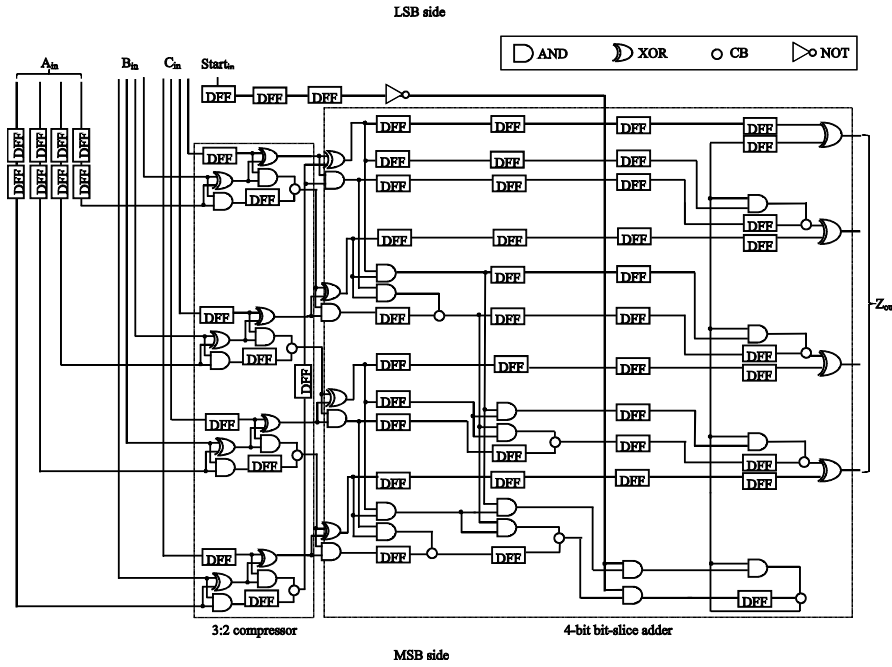


Fig. 5 A logic gate level circuit of the final addition cell.

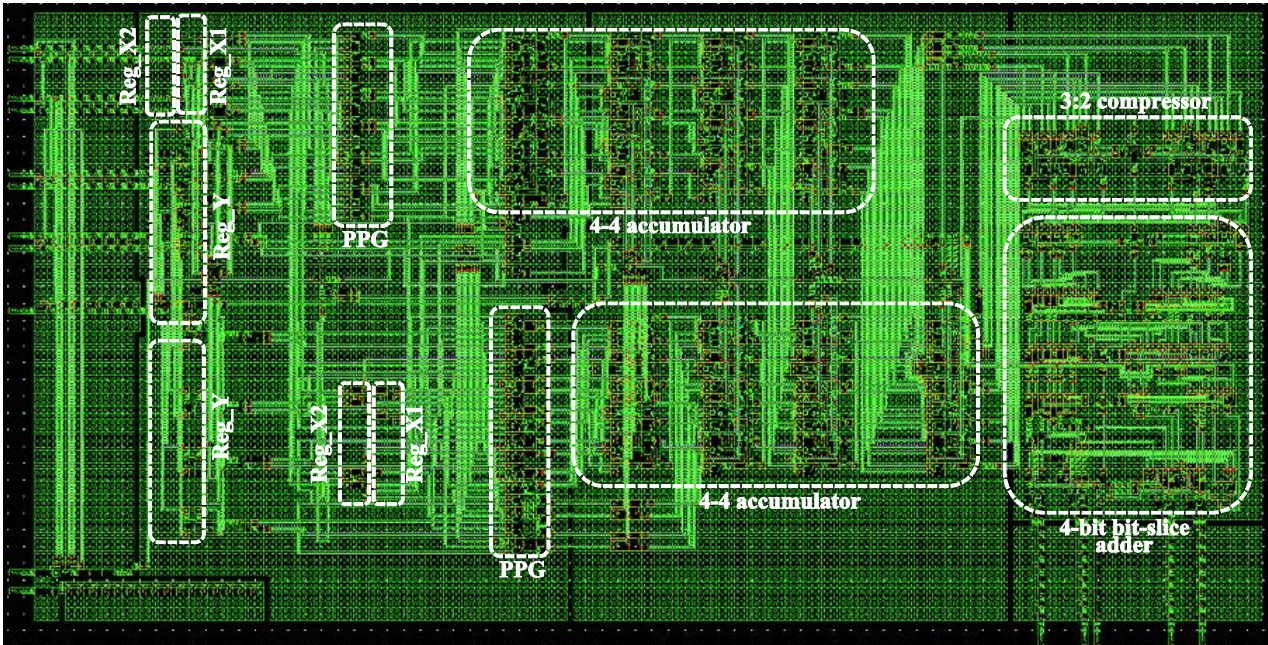


Fig. 6 The physical layout of an 8×8 -bit 4-bit bit-slice multiplier.

at the $(2n + 18)$ -th to $(4n + 17)$ -th clock cycles. The latency for a $4n \times 4n$ -bit multiplication is $4n + 17$ clock cycles (plus circuit delay). Namely, the most significant slice of the resultant product is output after $4n + 17$ clock cycles from the input of the least significant slice pair of operands. The multiplier can carry out a $4n \times 4n$ -bit multiplication every $2n$ clock cycles.

3. Layout and Simulation of an 8×8 -bit Multiplier

For verifying the algorithm and the logic design, a physical layout of an 8×8 -bit 4-bit bit-slice multiplier with SFQ-to-DC and DC-to-SFQ converters has been designed and simulated using the AIST ADP2 with target operating frequency of 50 GHz. Figure 6 shows the entire layout. The 8×8 -bit multiplier includes all the component blocks, i.e., the 4-4

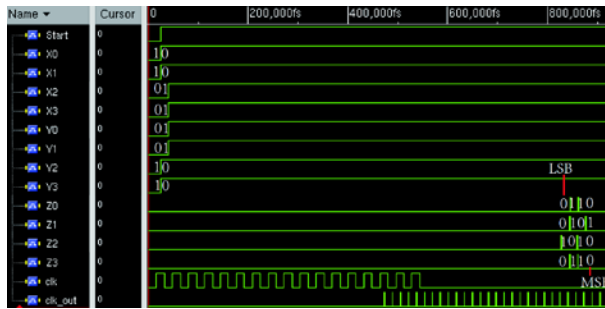


Fig. 7 A simulation result of unsigned multiplication $(1010\ 0101)_2 \times (0101\ 1010)_2 = (0011\ 1010\ 0000\ 0010)_2$ at 50 GHz.

accumulator, the PPG, the PPG for the most significant cell, the 3:2 compressor and the bit-slice adder.

In order to reduce the latency, PTLs are used for the wiring between stages. The 8×8 -bit 4-bit bit-slice multiplier consists of 21 stages and 11,488 JJs. It occupies the area of $5.3 \times 2.6\text{ mm}^2$. It has the bias current of 1,302 mA and the circuit delay of 460 ps at the bias voltage of 2.5 mV. Here, “circuit delay” is defined as the total delay of the logic gates from data-in to data-out, and has been estimated by static timing analysis. The latency is $20\text{ ps/cycle} \times 25\text{ cycles} + 460\text{ ps} = 960\text{ ps}$ at 50 GHz.

We have simulated the designed circuit with Cadence Verilog-XL software using the behavior model of the cells provided with the cell library. The simulation results show that multiplier operates correctly up to 62.5 GHz. Figure 7 shows a correct result of simulation.

4. Conclusion

An RSFQ 4-bit bit-slice multiplier based on a new systolic-like algorithm has been proposed. A logic design of a $4n \times 4n$ -bit multiplier using the AIST ADP2 includes $2n + 17$ stages. We have verified the algorithm and the logic design by making physical design of an 8×8 -bit multiplier.

Comparing with the parallel architecture, the bit-slice approach simplifies the circuit complexity and reduces the hardware cost. We believe that the bit-slice processing is a practical solution for a multiplication with longer operand length. Although we have let the length of a bit-slice be four in this paper, we can design a bit-slice multiplier with any slice length in the same way. When we design an $m \times m$ -bit k -bit bit-slice multiplier, it will mainly consist of m/k cells and the amount of hardware of each cell is proportional to k^2 . Therefore, the amount of hardware of the multiplier will be $O(km)$ instead of $O(m^2)$.

Acknowledgments

This work was partly supported by ALCA-JST in Japan. The authors thank Mr. Y. Kawaguchi, and Ms. Y. Ando of Kyoto University, and Prof. M. Tanaka, and Mr. R. Sato of Nagoya University for their assistance.

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, no.1, pp.3–28, March 1991.
- [2] N. Takagi and M. Tanaka, “Comparisons of synchronous-clocking SFQ adders,” *IEICE Trans. Electron.*, vol.E91-C, no.4, pp.429–434, April 2010.
- [3] A. Fujimaki, M. Tanaka, R. Kasagi, K. Takagi, M. Okada, Y. Hayakawa, K. Takata, H. Akaike, N. Yoshikawa, S. Nagasawa, K. Takagi, and N. Takagi, “Large-scale integrated circuit design based on a Nb nine-layer structure for reconfigurable data-path processors,” *IEICE Trans. Electron.*, vol.E97-C, no.3, pp.157–165, March 2014.
- [4] K. Takagi, M. Tanaka, S. Iwasaki, R. Kasagi, I. Kataeva, S. Nagasawa, T. Satoh, H. Akaike, and A. Fujimaki, “SFQ propagation properties in passive transmission lines based on a 10-Nb-layer structure,” *IEEE Trans. Appl. Supercond.*, vol.19, no.3, pp.617–620, June 2009.
- [5] I. Kataeva, H. Engseth, and A. Kidiyarova-Shevchenko, “New design of an RSFQ parallel multiply accumulate unit,” *Supercond. Sci. Technol.*, vol.19, pp.381–387, May 2006.
- [6] M. Dorojevets, A.K. Kasperek, N. Yoshikawa, and A. Fujimaki, “20-GHz 8×8 -bit Parallel Carry-Save Pipelined RSFQ Multiplier,” *IEEE Trans. Appl. Supercond.*, vol.23, no.3, Art. ID 1300104, June 2013.
- [7] K. Obata, M. Tanaka, Y. Tashiro, Y. Kamiya, N. Irie, K. Takagi, N. Takagi, A. Fujimaki, N. Yoshikawa, H. Terai, and S. Yorozu, “Single-flux-quantum integer multiplier with systolic array structure,” *Physica C*, vol.445–448, pp.1014–1019, 2006.
- [8] X. Peng, Q. Xu, T. Kato, Y. Yamanashi, N. Yoshikawa, A. Fujimaki, N. Takagi, K. Takagi, and M. Hidaka, “High-speed demonstration of bit-serial floating-point adders and multipliers using single-flux-quantum circuits,” *IEEE Trans. Appl. Supercond.*, vol.25, no.3, Art ID. 1301106, June 2015.
- [9] Y. Yamanashi, T. Kainuma, N. Yoshikawa, I. Kataeva, H. Akaike, A. Fujimaki, M. Tanaka, N. Takagi, S. Nagasawa, and M. Hidaka, “100 GHz demonstrations based on the single-flux-quantum cell library for the 10 kA/cm^2 Nb multi-layer process,” *IEICE Trans. Electron.*, vol.E93-C, no.4, pp.440–444, April 2010.
- [10] S. Nagasawa, K. Hinode, T. Satoh, M. Hidaka, H. Akaike, A. Fujimaki, N. Yoshikawa, K. Takagi, and N. Takagi, “Nb 9-layer fabrication process for superconducting large-scale SFQ circuits and its process evaluation,” *IEICE Trans. Electron.*, vol.E97-C, no.3, pp.157–165, March 2014.
- [11] G. Tang, K. Takata, M. Tanaka, A. Fujimaki, K. Takagi, and N. Takagi, “4-bit Bit-Slice Arithmetic Logic Unit for 32-bit RSFQ Microprocessors,” *IEEE Trans. Appl. Supercond.*, vol.26, no.1, Art. ID 1300106, Jan. 2016.



Guang-Ming Tang received the B.E. and M.E. degrees in information science from Guizhou University of Technology (now Guizhou University), Guiyang, China, in 2001 and Beijing University of Aeronautics & Astronautics (now Beihang University), Beijing, China, in 2009, respectively. He joined the Department of Computer Engineering, Guizhou University of Technology University of Technology, as an instructor from 2001 to 2010. His interests include computer architecture, hardware algorithms and logic design using RSFQ circuits. He is currently a Ph.D. candidate in Computer Engineering of Kyoto University, Kyoto, Japan.



Kazuyoshi Takagi received the B.E., M.E. and Dr. of Engineering degrees in information science from Kyoto University, Kyoto, Japan, in 1991, 1993 and 1999, respectively. From 1995 to 1999, he was a Research Associate at Nara Institute of Science and Technology. He had been an Assistant Professor since 1999 and promoted to an Associate Professor in 2006, at the Department of Information Engineering, Nagoya University, Nagoya, Japan. He moved to Department of Communications and Computer Engineering, Kyoto University in 2011. His current interests include system LSI design and design algorithms.



Naofumi Takagi received the B.E., M.E., and Ph.D. degrees in information science from Kyoto University, Kyoto, Japan, in 1981, 1983, and 1988, respectively. He joined Kyoto University as an instructor in 1984 and was promoted to an associate professor in 1991. He moved to Nagoya University, Nagoya, Japan, in 1994, and promoted to a professor in 1998. He returned to Kyoto University in 2010. His current interests include computer arithmetic, hardware algorithms, and logic design. He received Japan

IBM Science Award and Sakai Memorial Award of the Information Processing Society of Japan in 1995, and The Commendation for Science and Technology by the Minister of Education, Culture, Sports, Science and Technology in Japan in 2005.