

# 10 kA/cm<sup>2</sup> プロセスを用いた 2-bit ビットスライス・アダダーの設計と評価

高田 賢介<sup>1</sup> 早川 雄飛<sup>1</sup> 田中 雅光<sup>1</sup> 藤巻 朗<sup>1</sup>

1 名古屋大学大学院 工学研究科 〒464-8601 名古屋市千種区不老町

E-mail: takata@super.nuqe.nagoya-u.ac.jp

**あらまし** 現在、様々な研究機関で単一磁束量子 (Single Flux Quantum, SFQ) 論理回路を用いたデジタル回路の研究が行われている。その中で、我々はマイクロプロセッサの研究を行っている。これまで実証された SFQ マイクロプロセッサでは、ビットシリアル・アーキテクチャが用いられてきた。そこで、マイクロプロセッサのさらなる性能向上のため、ビットスライス・アーキテクチャの導入を念頭にビットスライス・アダダーの設計を行った。本稿では、ビットパラレル、シリアル、スライス・アーキテクチャを紹介し、それらに基づくアダダーの性能を演算時間と回路規模から比較し評価した。またビット・スライス幅が2ビットのビットスライス・アダダーを設計し、AIST 10 kA/cm<sup>2</sup> Nb アドバンスドプロセスを用いて試作を行い、高速試験において動作実証に成功したので報告する。

**キーワード** SFQ 論理回路, 10 kA/cm<sup>2</sup> プロセス, 2-bit ビットスライス・アダダー

## Design and Evaluation of the 2-bit Bit-Slice Adder Based on 10 kA/cm<sup>2</sup> Process

Kensuke TAKATA<sup>1</sup> Yuhi HAYAKAWA<sup>1</sup> Masamitsu TANAKA<sup>1</sup> Akira FUJIMAKI<sup>1</sup>

1 Department of Quantum Engineering, Nagoya University Furo-cho, Chikusa-ku, Nagoya, 464-8603 Japan

E-mail: takata@super.nuqe.nagoya-u.ac.jp

**Abstract** A large number of researches on designing digital circuits by using SFQ logic circuits have been undertaken extensively. We have been developing SFQ microprocessors. In our demonstrated SFQ microprocessors, bit serial architectures were used. In order to increase the performance of the microprocessors, we have designed a bit-slice adder toward introduction of a bit-slice architecture. In this paper, we present a bit-parallel, serial, and slice architectures, and we compare and evaluate the operating times and circuit sizes of the adders based on these architectures. We also report design of a 2-bit bit-slice adder using AIST 10 kA/cm<sup>2</sup> niobium advanced process, and demonstration of its high-speed operation.

**Keyword** SFQ circuit, 10 kA/cm<sup>2</sup> process, 2-bit bit-slice adder

### 1. はじめに

近年、大規模で複雑な科学技術計算などの分野においてスーパーコンピュータの需要が高まっている。これまでのコンピュータシステムの劇的な性能向上は半導体デバイスによる集積回路技術の飛躍的な進歩によって支えられてきた。しかし、デバイスの微細化には物理的な限界があり、消費電力、配線遅延、複雑さの問題等によって CMOS 集積回路の更なる性能の向上が危ぶまれている。

そこで、半導体に比べ1~2桁高いスループットと2~3桁低い低消費電力性を持つ単一磁束量子 (Single

Flux Quantum, SFQ) 回路 [1] が上記で述べたような CMOS 集積回路の問題を克服できる第一候補として期待されている。

我々は、SFQ 回路によるマイクロプロセッサの研究を行っている。これまでに動作実証された SFQ マイクロプロセッサ [2] では、設計の簡略化のために、1ビットずつ連続的にデータを処理していくビットシリアル・アーキテクチャが用いられてきた。しかし今後、プロセッサの高性能化を図るにあたって、より高速なデータ処理を可能にするために他のアーキテクチャの検討が必要となってくる。他のアーキテクチャとして

ビットパラレル・アーキテクチャが挙げられるが、回路が複雑で大規模であり、数十 GHz という高い周波数領域では配線遅延が無視できないため、タイミング調整が難化する。結果として動作周波数が低下し、プロセッサの性能に影響を与える。一方、データを数ビットずつ並列に処理するビットスライス・アーキテクチャは、回路の複雑化を緩和し、かつ SFQ 回路の高いスループットを生かしたアーキテクチャとして期待される。

これまでの SFQ マイクロプロセッサの動作実証では、臨界電流密度が  $2.5 \text{ kA/cm}^2$  の ISTECH Nb 標準プロセス 2 (STP2) [3] が用いられてきた。近年、SFQ 回路の更なる高性能化を目指して  $10 \text{ kA/cm}^2$  Nb アドバンスドプロセス 2 (ADP2) [4] が開発された。ADP2 は STP2 に比べ接合の臨界電流密度  $J_c$  が向上しており、設計上の指針においてもマッカンバー係数を 2 としたことから、回路の高速動作が可能となった。また STP2 の超伝導層が 4 層であるのに対し、ADP2 では 9 層に増加している。論理ゲートのレイアウトにおいても STP2 に比べ各辺の長さが  $3/4$  になるような設計をしているので、回路の高集積化が可能となり、回路面積の低減が期待できる。

そこで本論文では、各アーキテクチャに基づくアダーについて、演算時間と回路規模の 2 点から性能を比較する。また ADP2 を用いた SFQ-ビットスライス・プロセッサの 50 GHz での動作実証を最終目標とした上で、そのための最初の段階として、ADP2 を用いたスライス幅が 2 ビットのビットスライス・アダー (bit-slice adder, BSA) を設計し高速試験を行った結果について述べる。

## 2. アーキテクチャの比較

本稿では、データ長は 32 ビットとする。

### 2.1. ビットパラレル・アーキテクチャ

パラレル方式はデータを空間的に広げて並列処理する方式である。全データが回路に同時に入るため全データを入力するのに 1 クロックで済み、並列に計算処理ができるため、演算時間の短縮が見込める。しかし同時に処理するデータが多いため、回路規模は大きくなる。その概略図を図 1(a) に示す。

### 2.2. ビットシリアル・アーキテクチャ

シリアル方式は 1 ビットずつデータを処理していく方式である。そのため回路規模は小さくなる。しかし全データを入力するためのオーバーヘッドが大きく、並列に計算処理できないため演算時間の短縮が期待できない。その概略図を図 1(b) に示す。

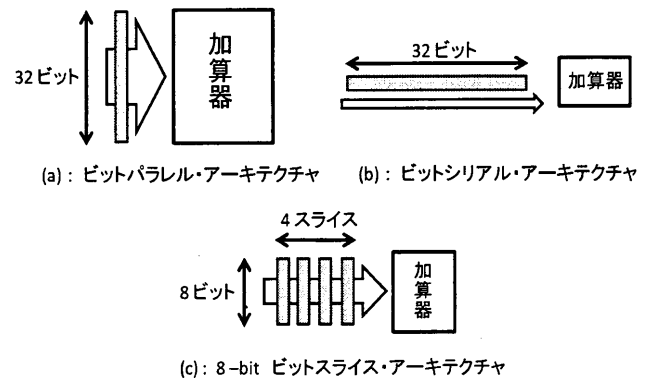


図 1 各アーキテクチャの概略図

### 2.3. ビットスライス・アーキテクチャ

スライス方式はパラレル方式とシリアル方式の両方の性質を併せ持つ方式である。ここでは分割されたデータをスライスと呼ぶことにする。この方式では、データをいくつか分割することで、データの並列処理が可能であり、データ入力に必要なクロック数も減らすことができる。したがってパラレル方式ほど回路規模が増大せず、加えて演算時間の短縮にもつながる。例えば、ビット・スライス幅を 8 ビットと仮定した場合、32 ビットのデータは 8 ビットのスライス 4 つに分けられる。回路に同時に入力できるスライスは 1 つであるため、全スライスを入力するのに 4 クロックが必要である。その概略図を図 1(c) に示す。

### 2.4 アダーの演算時間の算出方法

各アーキテクチャに基づくアダーの演算時間は次式から求めることができる。

$$(\text{演算時間}) = (\text{回路のレイテンシ}) + \left( \frac{\text{データ長}}{\text{ビット・スライス幅}} \right) \times \frac{1}{\text{動作周波数}}$$

回路のレイテンシとはデータが入力されてからそのデータが出力されるまでにかかる時間である。第二項は全データの入力にかかる時間であり、SFQ 論理回路の高い動作周波数を用いることで非常に小さくすることができる。ただしパラレルアーキテクチャではデータ長 = ビット・スライス幅であるため、(演算時間) = (回路のレイテンシ) となる。またビットシリアル・アダーではビット・スライス幅を 1 ビットとする。

## 3. アダーの設計

$n$  ビットの整数を入力とし  $n + 1$  ビットの和を求める加算を考える。アダーの入力は 2 入力とする。入力、出力については

入力： 加数  $A = a_{n-1} \cdots a_0$   
 被加数  $B = b_{n-1} \cdots b_0$  (\*)  
 出力： 和  $S = s_n s_{n-1} \cdots s_0$   
 と定義する。

アダーは ALU (arithmetic logic unit: 算術論理演算ユニット) の最も基本となる回路である。マイクロプロセッサにおいて加算命令は最も頻度の高い算術命令であり、また減算や比較を行う回路も基本的な構成はアダーに基づいている。したがって、アダーの性能を定量的に見積もることでプロセッサの演算能力を予測することができる。

### 3.1 加算アルゴリズム [5]

加算は、各桁においてその桁の加数  $a_i$ 、被加数  $b_i$  と下位からの桁上げ  $c_i$  の排他的論理和によりその桁の和  $s_i$  が求められ、その和  $s_i$  と上位への桁上げ  $c_{i+1}$  を下位の桁から順に計算していくことで行える。このとき  $s_i$ 、 $c_{i+1}$  は次式で表される。

$$s_i = a_i \oplus b_i \oplus c_i$$

$$c_{i+1} = a_i \cdot b_i + (a_i \oplus b_i) \cdot c_i$$

ここで、 $\cdot$  は論理積、 $+$  は論理和、 $\oplus$  は排他的論理和を表す。この 1 桁分の計算を行う回路を全加算器 (full adder, FA) と呼ぶ。全加算器の回路図を図 2 示す。

$n$  ビットの加算において図 3 に示すように  $n$  個の全加算器を接続した構造をとるのが順次桁上げ加算器 (ripple carry adder, RCA) である。回路の構成を図 3 に示す。RCA では、 $n$  ビットの加算を行うには、桁上げが最下位から最上位まで伝搬しなければならない構造であるため、演算時間は  $O(n)$  になる。

そこで演算時間をより短くするために考案されたものが桁上げ先見加算器 (carry look-ahead adder, CLA) であり、高速なアルゴリズムとして知られている。

CLA の原理を考えるために第  $i$  桁において

$$g_i = a_i \cdot b_i, p_i = a_i \oplus b_i$$

とおく。 $g_i$  は桁上げ発生信号と呼ばれ、1 のときこの桁で桁上げが発生 (generate) することを意味する。 $p_i$  は桁上げ伝搬信号と呼ばれ、1 のとき下位から桁上げが上位に伝搬 (propagate) することを意味する。

第  $i$  桁および第  $i+1$  桁の 2 桁からなるブロックの桁上げ発生条件と伝搬条件をそれぞれ  $g_{i+1,i}$ 、 $p_{i+1,i}$  とするとこれらは次の計算で求められる。

$$g_{i+1,i} = g_{i+1} + p_{i+1} \cdot g_i$$

$$p_{i+1,i} = p_{i+1} \cdot p_i$$

この計算を演算子  $\circ$  で定義する。

$$(g_{i+1,i}, p_{i+1,i}) = (g_{i+1}, p_{i+1}) \circ (g_i, p_i)$$

・演算子には結合律が成立するため、第  $i$  桁から第  $j$  桁 ( $j \geq i$ ) までのブロックの桁上げ生成条件と伝搬

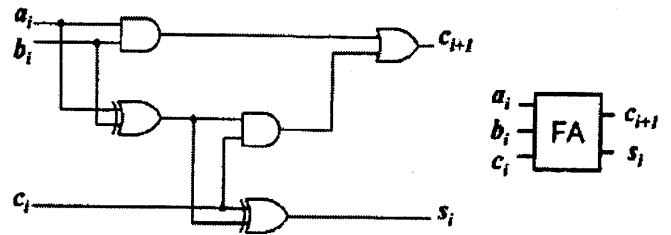


図 2 全加算器 (FA) の回路図

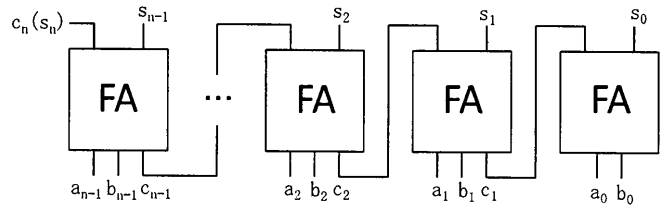


図 3 順次桁上げ加算器 (RCA) の構成

条件  $g_{j,i}$ 、 $p_{j,i}$  は次のように表すことができる。

$$(g_{j,i}, p_{j,i}) = (g_j, p_j) \circ \cdots \circ (g_{i+1}, p_{i+1}) \circ (g_i, p_i)$$

$g_{i,0}$  および  $p_{i,0}$  が求めれば、桁上げ  $c_{i+1}$  は次の式で計算できる。

$$c_{i+1} = g_{i,0}$$

また下位からの桁上げ  $c_i$  が求めれば、和は

$$s_i = g_i \oplus c_i$$

により計算される。

桁上げを並列に計算することにより演算時間は  $O(\log n)$  に減少する。しかし回路の素子数は  $O(n \log n) \sim O(n^2)$  まで増加する。演算子  $\circ$  を並列に組み合わせる順序を工夫することにより、様々な CLA の構成が提案されている。その中で演算子  $\circ$  の段数が最も少ない 8-bit Kogge-Stone CLA [6] の構成を図 4 に示す。

### 3.2 ビットスライス・アダー (BSA) [7]

2.1 や 3.1 で述べたように、ビットパラレル・アーキテクチャである CLA は演算時間の短縮は期待できるが、素子数が増大してしまう。そのため演算時間と素子数の両面から評価するとき、バランスがとれた回路とは言えない。そこで CLA の加算アルゴリズムを導入した BSA を考える。ビット・スライス幅が 2 ビットの場合、(\*) の加算において、加数 A、被加数 B はそれぞれ

$$A_0 : a_{n-2} a_{n-4} \cdots a_0 \text{ と } A_1 : a_{n-1} a_{n-3} \cdots a_1$$

$$B_0 : b_{n-2} b_{n-4} \cdots b_0 \text{ と } B_1 : b_{n-1} b_{n-3} \cdots b_1$$

に分割され、出力 S は

$$S_0 : s_n s_{n-2} \cdots s_0 \text{ と } S_1 : s_{n-1} s_{n-3} \cdots s_1$$

に分割される。この BSA の特徴としては、CLA のようにデータを並列処理することにより高速演算を可能にし、かつ回路を構成する素子数の増大を抑えられるこ

- (G,P)** : g(生成)とp(伝搬) 信号生成回路
- : ◦ 演算回路
- DFF** : 桁上げ信号とg信号の同期をとるためのDFF回路
- ⌋** : 和を計算するための排他的論理和回路

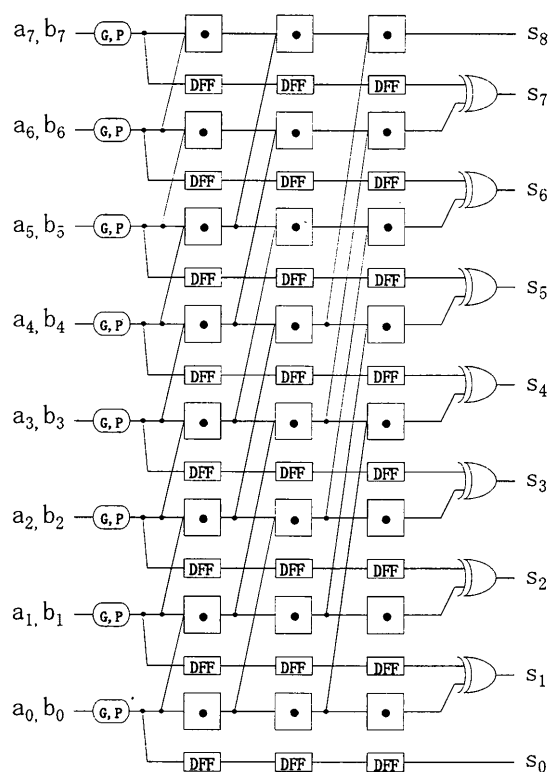
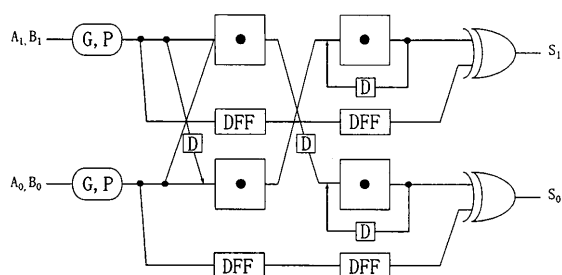


図4 8-bit Kogge-Stone CLA の構成



- D** : 次のスライスへキャリーを渡すためのDFF

図5 2bit-BSA の構成図

とである。ただし、データは同時に全て入力されるわけではなく、クロック毎にスライスが数回に分けられて入力されるため、前後のスライス間に存在するつながり情報（キャリー情報）を次に渡す回路が必要になる。我々はこの回路にDフリップフロップ(DFF)を用いる。図5に8-bit Kogge-Stone CLAのアルゴリズムに基づくBSAの例として、ビット・スライス幅が2ビットのBSAの構成を示す。

#### 4. スライスアダーとシリアルアダーの定量的な比較と評価

加算において、データ長やビット・スライス幅が変われば、データの入力に必要なクロック数と回路のレイテンシが変わり、演算時間に影響を与える。また回路規模（ジョセフソン接合の数や回路面積）についてもビット・スライス幅に依存する。そこでビットシリアル・アダーとスライス幅が異なるビットスライス・アダーの詳細設計を行い、演算時間、回路規模の2点を定量的に比較し、評価する。ビットパラレル・アダーに関しては回路規模が非常に大きく、十分な検討ができなかったため比較から除外した。

以下で比較を行う4つのアダーは、詳細設計をしたビットシリアル・アダー、2-bit、4-bit BSA と、4-bit BSAをもとに見積もった8-bit BSAである。データ長は32ビットの場合を考える。

##### 4.1 演算時間の比較と評価

図6にビットシリアル・アダーと2-bit、4-bit、8-bit BSAの演算時間の動作周波数依存性を示す。

このグラフから、データ長が32ビットの場合には、ビットシリアル・アダーは全データを入力するためのオーバーヘッドが大きいため演算処理に時間がかかってしまうことが分かる。BSAはそのオーバーヘッドを抑え、データの並列処理が可能であるために演算時間に優位性があると言える。さらに目標としている50GHzでは4-bit BSAが最も演算時間が短いことが分かる。

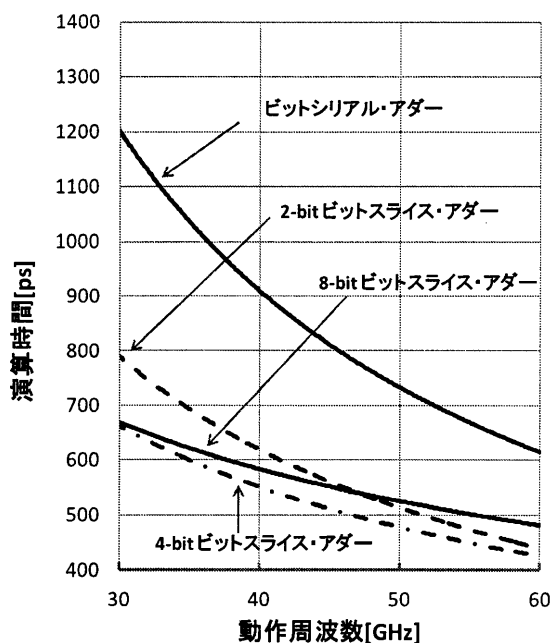


図6 ビットスライス・アダーとビットシリアル・アダーの演算時間の動作周波数依存性

### 4.2 回路規模の比較と評価

表 1 にビットシリアル・アダーと 2-bit、4-bit、8-bit BSA におけるジョセフソン接合の数と回路面積を示す。

この表から、ビット・スライス幅が小さければ、回路規模が小さくなることは明らかである。4-bit BSA はまだ設計の段階で改善の余地があると考えられ、今後より回路規模を小さくできる可能性がある。4-bit BSA から回路規模を見積もりで算出した 8-bit BSA に関しても同じことが言える。

### 4.3 演算時間と回路規模の両面からの比較と評価

図 7 に接合数の演算時間依存性を示す。ただし演算時間は動作周波数が 50 GHz の時の値である。

図 7 からデータ長が 32 ビットの場合、2-bit BSA や 4-bit BSA が他のアダーに比べ演算時間と回路規模とのバランスに優位性があると考えられる。4-bit BSA の回路規模の改善とそれに伴う演算時間の短縮を考えると、データ長が 32 ビットの場合、4-bit BSA が最もバランスのとれたアダーである考えられる。

## 5. 2-bit ビットスライス・アダーの試作と動作実証

今回はデータ長を 8 ビットとして、ADP2 による 2-bit BSA の詳細設計し、実験による評価を行った。ただし試作した 2-bit BSA は 4 節で述べた 2-bit BSA よりも前に設計した回路であるため、演算時間や接合数、回路面積が 4 節での値はやや大きくなっている。

設計には CONNECT セルライブラリ [8] を用いた。また設計上の動作目標周波数は 50 GHz である。チップ外部からこれほどの高速なデータを入力し、高速な出力を観測することは難しいため、オンチップテストによる高速試験 [9] を行った。高速テスト用の回路は 2-bit BSA、ラダー型クロック発生器 (clock generator: CG)、高速入力用シフトレジスタ (SR\_IN)、高速出力用シフトレジスタ (SR\_OUT) で構成されている。図 8 に 2-bit BSA のテスト回路全体の顕微鏡写真を示す。

テスト用の回路を除く、2-bit BSA のみのジョセフソン接合の数は 1100、面積は 2.44 mm×0.90 mm である。またロジックシミュレーションから得られた演算時間は 561 ps であった。

次に測定結果について述べる。例として、加数、被加数をそれぞれ、A(11111011)、B(01111110) とする。この時  $A_0(1101)$ 、 $A_1(1111)$ 、 $B_0(1110)$ 、 $B_1(0111)$  となる。和 S は (101111001) となる。図 9 に出力波形を示す。データは下位の桁から上位の桁への順となっている。出力は  $S_0(11101)$ 、 $S1(0110)$  となっており、正しい加算結果が得られている。

表 1: 接合数と回路面積

|           | 接合数  | 回路面積            |
|-----------|------|-----------------|
| シリアルアダー   | 240  | 0.53 mm×0.25 mm |
| 2-bit BSA | 910  | 1.10 mm×0.53 mm |
| 4-bit BSA | 2670 | 3.26 mm×1.01 mm |
| 8-bit BSA | 6850 | 4.13 mm×2.23 mm |

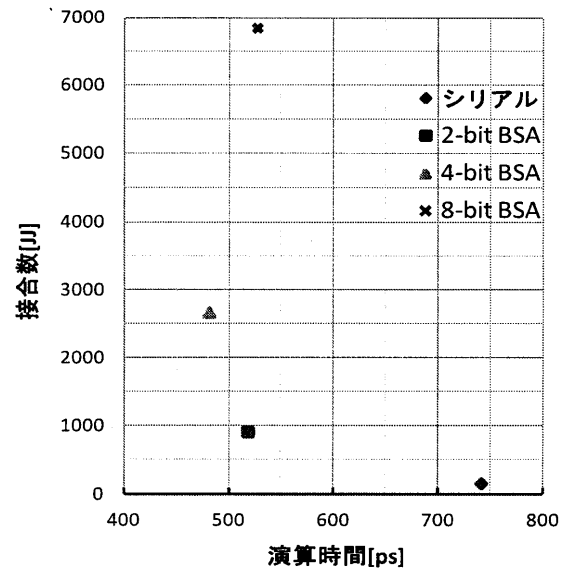


図 7 接合数の演算時間依存性

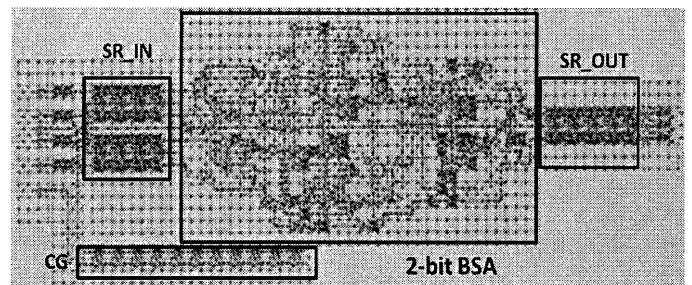


図 8 2-bit BSA の顕微鏡写真

次に 2-bit BSA の DC バイアスマージンの動作周波数依存性を図 10 に示す。実線で囲まれた部分が動作領域である。DC バイアスは設計値を 100% とした。ただし今回、設計のミスにより動作周波数が低い領域ではタイミングエラーが起きるため、シミュレーションと測定値ともに高周波数領域のみを示す。

図 10 からシミュレーションに比べ測定値の動作領域が狭く、高バイアス、高周波数側にシフトしていることが分かる。この原因として、今回の試作では臨界電流値  $I_c$  が設計値より高くなっていることが考えられる。 $I_c$  の定量的な評価はまだ行っていない。図 10 に示した、測定値における動作周波数は、設計値の  $I_c$  を仮定してシミュレーションを行った結果をもとに算

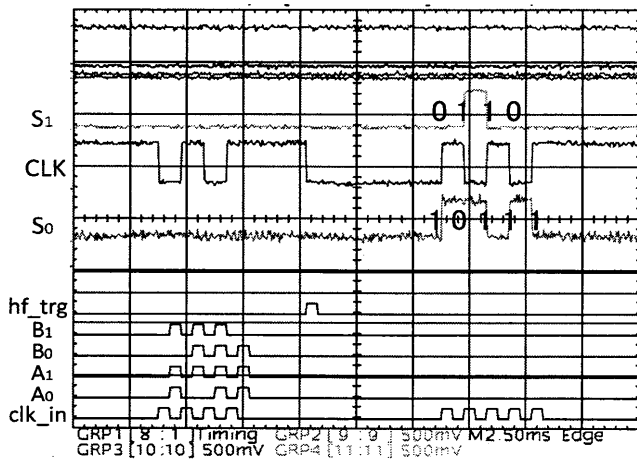


図 9 2-bit ビットスライス・アダーの出力波形

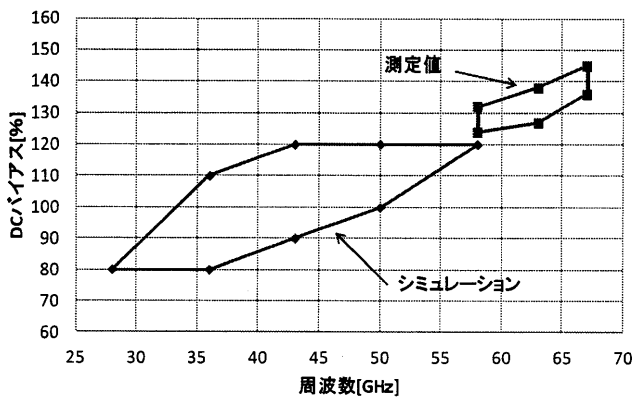


図 10 2-bit BSA の DC バイアスマージンの動作周波数特性

出した値である。したがって、実測に基づき  $I_c$  を高く設定したシミュレーション結果をもとに動作周波数を算出すれば、実際の動作周波数は図 10 に示した値よりも低くなると考えられる。

## 6. まとめ

データ長を 32 ビットとし、ビットシリアル、スライス・アーキテクチャに基づくアダーを演算時間と回路規模の 2 点から比較し、マイクロプロセッサの高性能化を図るために適したアダーの検討を行った。その結果、2-bit、4-bit BSA が演算時間と回路規模のバランスがとれていた。さらにデータ長を 8 ビットとした ADP2 に基づく 2-bit ビットスライス・アダーを設計し高速動作の実証に成功した。

## 謝辞

本研究は、ALCA-JST の支援を受けたものである。本研究に使用された回路は、(独) 産業技術総合研究所 (AIST) の超伝導クリーンルームにおいて、AIST-ADP2 プロセスを用いて作製された。AIST-ADP2 は、(公財)

国際超電導産業技術研究センター (ISTEC) のニオブ集積回路プロセスを AIST にて発展させたものである。

## 参考文献

- [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, March 1993.
- [2] A. Fujimaki, M. Tanaka, T. Yamada, Y. Yamanashi, H. Park, N. Yoshikawa, "Bit-Serial Single Flux Quantum Microprocessor CORE," *IEICE Trans. Electron.*, vol. E91-C, no. 3, pp. 342-349, March 2008.
- [3] S. Nagasawa, Y. Hashimoto, H. Numata and S. Tahara, "A 380 ps, 9.5 mW Josephson 4-Kbit RAM operated at a high bit yield", *IEEE Trans. Appl. Supercond.*, vol.5, pp.2447-2452, Jun 1995.
- [4] S. Nagasawa, et al, "New Nb Multi-layer Fabrication Process for Large-Scale SFQ Circuits," *Physica C*, vol. 469, pp. 1578-1584, Oct. 2009.
- [5] 高木直史, 算術演算の VLSI アルゴリズム, 萩原宏 (編), pp.13-26, コロナ社, 東京, 2005.
- [6] P. M. Kogge and H. S. Stone, "A Parallel Algorithm for the efficient Solution of a General Class of Recurrence Equations," *IEEE Trans. Computers*, vol.C-22, pp. 786-792, August 1973.
- [7] 朴熙中, SFQ 論理回路を用いた 4-b ビット・スライス・アダーの設計, 超伝導エレクトロニクス研究会 2007
- [8] 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<sup>2</sup> Nb Multi-Layer Process," *IEICE Trans. Electron.*, vol. E93-C, no. 4, pp. 440-444, Apr. 2010.
- [9] Z. J. Deng, N. Yoshikawa, S. R. Whiteley and T. Vanduzer, "Data-Driven Self-Timed RSFQ Digital Integrated Circuit and System," *IEEE Trans. Appl. Supercond.*, vol. 7, pp. 3634-3637, June 1997.