

単一磁束量子回路向け マイクロプロセッサのアーキテクチャ探索

石田 浩貴^{1,a)} 田中 雅光² 小野 貴継¹ 井上 弘士¹

受付日 2016年5月31日, 採録日 2016年12月1日

概要: 近年, CMOS マイクロプロセッサは消費電力の問題のため動作周波数の向上が頭打ちとなっている. この問題に対し, 我々は CMOS 集積回路が直面している消費電力問題を根本的に解決する特徴を原理的に有している単一磁束量子 (SFQ: Single-Flux-Quantum) 回路を用いた超高性能マイクロプロセッサに着目した. 本論文では, SFQ マイクロプロセッサの実現に向け, それに適したアーキテクチャを検討した. 得られた検討結果に基づき SFQ マイクロプロセッサを論理設計し, 性能および消費電力を見積もり CMOS マイクロプロセッサと定量的に比較した. その結果, 設計した SFQ マイクロプロセッサは, 最大で CMOS マイクロプロセッサ・モデルの約 35 倍の性能を 1.46 W という低消費電力で達成する可能性があることが分かった. また, 回路面積を SFQ に適したアーキテクチャの実装前と比較したところ, 3.58 倍になることが明らかになった.

キーワード: マイクロプロセッサ, 単一磁束量子, SFQ 回路, ジョセフソン接合, 低消費電力, 高性能

Exploring Design Space of a Single-flux-quantum Microprocessor

KOKI ISHIDA^{1,a)} MASAMITSU TANAKA² TAKATSUGU ONO¹ KOJI INOUE¹

Received: May 31, 2016, Accepted: December 1, 2016

Abstract: CMOS microprocessors have been facing a limitation for clock speed improvement because of increasing computing power. Using SFQ (Single-Flux-Quantum) devices and circuits is a promising way to solve the power wall problem. In this paper, we explore architecture design space of SFQ microprocessors to exploit SFQ potential. In addition, we estimate performance, area, and power consumption of the proposed SFQ microprocessor based on gate-level logic design. The results show that the SFQ microprocessor is able to outperform a CMOS microprocessor model by 35X with 1.46 W. Moreover, the area of the SFQ microprocessor is 3.58X larger than a baseline SFQ microprocessor.

Keywords: Microprocessors, Single flux quantum, SFQ circuits, Josephson junctions, Low power, High performance

1. はじめに

現代の CMOS マイクロプロセッサでは, 消費電力の増加により動作周波数の向上は困難である. この問題に対処すべく, チップ上に搭載した複数コアによるオンチップ並列処理が導入されている. しかしながら, 並列化が困難なプログラム部分の実行時間はシングルスレッド性能に依存す

るため, 依然として動作周波数の向上に対する要望は高い.

今後, マイクロプロセッサの性能を飛躍的に向上させるためには, CMOS に代わる新しいデバイスが強く求められる. 本論文では単一磁束量子 (SFQ: Single-Flux-Quantum) 回路に着目する. SFQ 回路は, 超伝導ループ間を磁束が移動する際に発生する微弱電圧パルスを用いて論理演算を行う集積回路である. その動作原理上, 電荷の充放電を必要としないため, 論理ゲートあたりの消費電力は CMOS 回路と比べて約 1/1,000 以下という性質を持つ [9], [18]. また, 超伝導状態により抵抗成分が限りなくゼロに近くなるため, 高速なパルス伝搬が可能という特性も有する. このような

¹ 九州大学
Kyushu University, 819-0395, Fukuoka

² 名古屋大学
Nagoya University, Nagoya 464-8603, Aichi

^{a)} koki.ishida@cpc.ait.kyushu-u.ac.jp

特徴に着目し、これまで SFQ マイクロプロセッサの試作・研究が進められている [11], [12], [13], [16]. 実際, SFQ マイクロプロセッサの CORE 1 β では, 25 GHz での動作実証に成功している [11], [16]. しかしながら, 25 GHz で動作するのはマイクロプロセッサ内の各論理ゲートであり, マイクロプロセッサ全体の動作周波数は 1.5 GHz にとどまる. これは SFQ 回路の特徴を活かすようなマイクロアーキテクチャの考慮が不十分であり, 回路面積の縮小やタイミング設計の容易化を優先とするデータ処理方式や, 動作原理の異なる CMOS マイクロプロセッサを前提としたパイプライン構成をとっていることが原因である.

本論文では, SFQ マイクロプロセッサに適したアーキテクチャを検討し, 性能, 消費電力, および, 面積を評価する. まず, 性能モデルを用いて SFQ 回路に適したマイクロアーキテクチャを検討した. 検討の結果, SFQ 回路の特性を考慮して高性能な動作を実現するためには, ビットパラレル方式と深いパイプライン構成が必要であることが明らかになった. そこで, この検討によって得られた結果に基づき, SFQ マイクロプロセッサの論理設計を行い, 性能および消費電力を評価した. その結果, 動作周波数は 222.22 GHz を達成可能であり, 性能は商用 CMOS マイクロプロセッサの構成を参考にした従来型モデルと比較して最大約 35 倍, 消費電力は 1.46 W を達成できる可能性があることが分かった. 一方, SFQ 回路の面積は提案アーキテクチャの実装前と比較して 3.58 倍になるという課題が明らかになった. これはレジスタファイルが増加したことが原因である.

本論文の構成は以下のとおりである. 2 章では関連研究について述べる. 3 章では, SFQ マイクロプロセッサに用いられている SFQ 回路の動作原理, および特徴について説明する. 4 章では, SFQ マイクロプロセッサに適したアーキテクチャの検討を行う. 具体的には, SFQ マイクロプロセッサのアーキテクチャ設計空間を整理し, 性能モデルを用いて評価を行う. そして今後とるべきアーキテクチャ設計指針を示す. 5 章では, 4 章で述べたアーキテクチャ設計指針によって期待できる性能やオーバーヘッドを明らかにするために SFQ マイクロプロセッサを論理設計する. 6 章では, 論理設計した SFQ マイクロプロセッサの性能, 消費電力, および, 面積を評価し, 提案アーキテクチャの有効性ならびに SFQ マイクロプロセッサの可能性を示す. 最後に 7 章でまとめる.

2. 関連研究

SFQ 回路は, 超伝導体ジョセフソン接合を用いた次世代集積回路の 1 つである. ジョセフソン接合は半導体を超える高速性と低消費電力性を兼ね備えたデバイスである^{*1}.

^{*1} ジョセフソン接合の詳細に関しては 3 章で述べる.

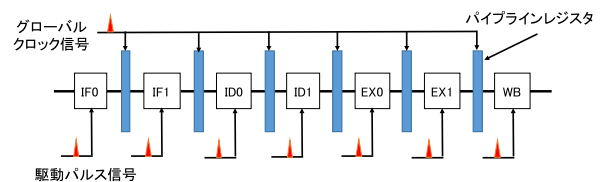


図 1 CORE 1 β のパイプライン構造
Fig. 1 Pipeline structure of CORE 1 β .

このような特徴に着目し, ジョセフソン接合を基本デバイスとする SFQ 回路を用いたマイクロプロセッサに関する研究が進められている [11], [12], [13], [16]. なかでも代表的なものとして CORE 1 β があげられ, 25 GHz の動作実証に成功している [11], [16]. これらの設計事例では, 回路面積の縮小やタイミング設計の簡素化, および, 回路動作の高速化を優先していることから, データ処理方式としてビットシリアル方式を採用している. ビットシリアル方式とは, データ語長 (マイクロプロセッサが扱うことができるデータのビット幅) を 1 ビット単位で処理 (たとえば, 32 ビット語長の場合には 1 ビット処理を 32 回繰り返す) する方式である. ビットシリアル方式を採用することで, データ語長に比例して処理回数が増加することから実行時間が長くなるという問題が生じる.

CORE 1 β は図 1 のように 7 段のパイプラインで構成されている. 各パイプラインはグローバルクロックで, 各論理ゲートは駆動パルス信号で同期されている. パイプラインステージ内の最長データパス (最も多くの論理ゲートが含まれるデータパス) によって動作周波数は決定するため, 駆動パルス信号が 25 GHz と高速動作をしているにもかかわらずグローバルクロック周波数は 1.5 GHz にとどまっているのが現状である.

このように, SFQ 回路は CMOS 回路を凌駕する可能性があるものの, これまでに検討されている SFQ マイクロプロセッサは高い性能を達成できていない. この原因として, SFQ 回路は CMOS 回路とまったく異なる動作原理や特徴を有しているにもかかわらず, その特徴を考慮し最大限に活用するマイクロアーキテクチャを採用していないことがあげられる. SFQ 回路を用いた高性能かつ低電力なプロセッサの実現に向けて, SFQ 回路に適したマイクロアーキテクチャの探索が不可欠である.

3. SFQ 回路の動作原理

3.1 ジョセフソン接合

ジョセフソン接合は SFQ 回路の基本デバイスであり, 図 2(a) のように 2 つの超伝導体の間に薄い障壁層をはさみ構造を有する [1]. 本論文では超伝導体としてニオブ (Nb), 障壁層にアルミ酸化物 (AlO_x) を用いた超伝導体-絶縁体-超伝導体型のジョセフソン接合を対象とする.

図 2(b) にジョセフソン接合の電気的特性を示す. ジョ

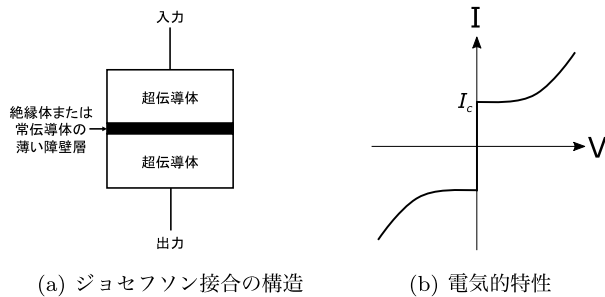


図 2 ジョセフソン接合の構造と電気的特性

Fig. 2 Structure and electric characteristics of Josephson junctions.

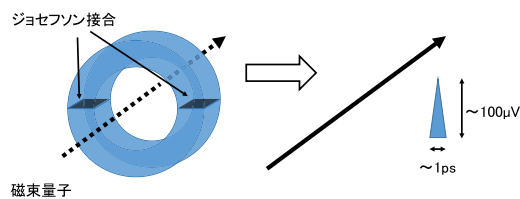


図 3 ジョセフソン接合を含む超伝導ループ

Fig. 3 Superconducting loop including Josephson junctions.

セフソン接合を流れる電流が臨界電流値 I_c を下回ると、ジョセフソン接合間には電圧が発生せず超伝導体間を直流電流が流れる。一方、 I_c を上回ると、ジョセフソン接合間に電位差が発生する。この電圧が発生する状態になることをスイッチする（あるいは接合が切れる）という。

3.2 スイッチング動作

SFQ 回路は、図 3 に示すようにジョセフソン接合を含む超伝導体のループによって構成されている。磁束は超伝導体のループ内で $\Phi_0 = h/2e = 2.07 \times 10^{-15} \text{ Wb}$ の整数倍に量子化され (h はプランク定数, e は電子の電荷), SFQ 回路ではこの磁束量子 Φ_0 を情報担体として用いる。超伝導体のループ内に磁束が存在する/しない場合をそれぞれ '1'/'0' のビット情報を保持していると見なす。たとえば、図 3 のようにループ内に磁束が入った状態を考えると、ループ部分には手前から見て時計回りの方向に周回電流が流れる。ループ 1 周のインダクタンスを L とすると、 $I_L \approx \Phi_0/L$ の大きさの周回電流が流れ、 I_L が I_c を下回る場合には磁束量子がループ内にとどまる。一方、 I_L が I_c より大きい場合はジョセフソン接合がスイッチし、ループ内の磁束量子はループ外に出る。ジョセフソン接合は、スイッチした後ループ内の磁束量子がなくなるとただちに電圧 0 の状態に戻る。

3.3 パルス論理

ジョセフソン接合がスイッチするとき、接合の両端にはインパルス状の電圧 (SFQ パルスと呼ばれる) が発生する。SFQ 回路では、この SFQ パルスの有無で '0' と '1' を表すパルス論理を用いて情報を表現する。CMOS 回路で

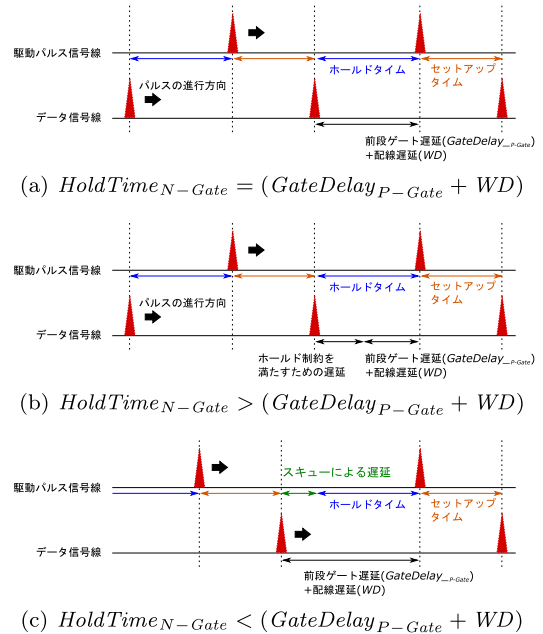


図 4 スキューの挿入条件

Fig. 4 Conditions of inserting skew elements.

用いられているレベル論理とは異なり、パルス自体では状態を表現できないため、'0' という情報を扱うための工夫が必要となる。たとえば、'1' という情報をパルスの到着で表すとする、'0' とパルスが到着していない状態との区別ができないためである。そこで、駆動パルス信号と呼ばれる論理ゲートごとにデータの同期を行う信号を用意し、一定間隔で到着する駆動パルス信号の 1 周期内にデータパルスが論理ゲートに到着すれば '1'、到着しなければ '0' と判定する。

SFQ マイクロプロセッサでは、パイプライン全体の動作を制御するグローバルクロック信号、ならびに前述の駆動パルス信号の 2 種類のクロック信号が存在する。グローバルクロック信号の周波数はクリティカルパスを含むパイプラインステージの遅延時間によって決定される。一方、駆動パルス信号の周波数は以下の SFQ 回路で構成される論理ゲートのホールドタイム、ならびにセットアップタイムによって決定される。

- セットアップタイム：データパルス信号入力後、次の駆動パルス信号入力までに空けておくべき時間
- ホールドタイム：駆動パルス信号入力後、データパルス信号入力までに空けておくべき時間

SFQ 回路では、駆動パルス信号周波数を最大にするためにスキューと呼ばれる回路遅延要素を用いる。スキューの挿入条件を図 4 に示す。ここで $HoldTime_{N-Gate}$ は次段論理ゲートのホールドタイム、 $GateDelay_{P-Gate}$ は前段論理ゲートの遅延、 WD は配線遅延を表している。図 4(a) は $HoldTime_{N-Gate} = (GateDelay_{P-Gate} + WD)$ の場合、図 4(b) は $HoldTime_{N-Gate} > (GateDelay_{P-Gate} + WD)$ の場合を示している。図 4(a), (b) の場合はスキューを

挿入することなく実行を続けることが可能である。一方、 $HoldTime_{N-Gate} < (GateDelay_{P-Gate} + WD)$ の場合、図 4(c) のように駆動パルス信号線にスキューを挿入し、駆動パルス信号の到着を遅らせる。これにより、駆動パルス信号の周波数を落とすことなく実行を続けることが可能である。

4. アーキテクチャ探索

4.1 アーキテクチャパラメータ

ここでは、命令パイプライン構造をとる場合に考えられる 2 つのアーキテクチャパラメータについて述べる。

- **パイプライン深度**：パイプライン深度はパイプラインステージ数を表す。CMOS/SFQ のいずれの回路を前提とした場合でも、マイクロプロセッサの性能を決める重要な要因となる。基準として、一般的な 5 段の命令パイプラインを考える。これをより深くしたパイプラインを持つものはスーパーパイプラインと呼ばれ、1 パイプラインステージの遅延が小さくなるために動作周波数を向上させることができる。また、究極には論理ゲートレベルまでパイプラインを深くしたゲートレベルパイプラインも考えられる。パイプライン深度を深くすることで動作周波数の向上を図れるが、1) パイプラインレジスタの面積オーバーヘッド、2) 動作周波数向上による消費電力の増加、といった欠点が存在する。しかしながら、SFQ 論理ゲートはラッチ機能を有しているため、パイプラインを深くするためにパイプラインレジスタを追加する必要がない。また、SFQ 論理ゲートは CMOS の場合と比較して $1/1,000$ 以下の消費電力で動作可能である。これらの原理的特徴を活用することで、上記 2 つの問題の解決が可能となる。したがって、パイプライン深度を深くすることによる性能向上がおおいに期待できる。
- **データバスビット幅**：データバスビット幅もまたアーキテクチャパラメータの 1 つであり、ALU やレジスタファイルなどの各ユニットが 1 度の処理で扱うビット幅を表す。データ語長は、マイクロプロセッサが扱うことができるデータのビット幅である。スライスとは、データ語長を分割したデータを指す。たとえば、64 ビットを 8 分割して得られた 8 ビットの各データをスライスと呼ぶ。また、この場合スライス数は 8 となる。以上をふまえ、パイプライン深度と同様に設計空間の定義を行う。ビットシリアル/スライス方式では、一定の駆動パルス信号間隔ごとに分割されたデータを投入し、SFQ 論理ゲートがそのラッチ機能を用いてデータを保持することでユニット内でのパイプライン化が可能となる。図 5 にビットシリアル/スライス方式でのデータ処理について示す。ここでは、論理ゲート 8 段、1 つのデータを 4 つのスライスに分割した場

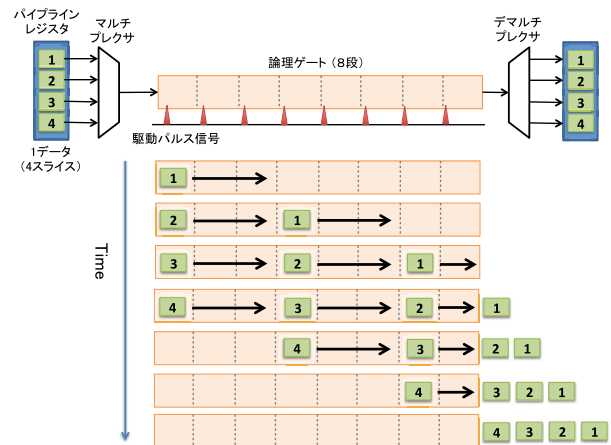


図 5 ビットシリアル/スライス方式におけるスライスのオーバーラップ実行

Fig. 5 Overlap execution in bit-serial/slice processings.

合を想定しており、Time は時間経過を表す。この図からも分かるように、同一ユニット内に複数スライスが存在可能になり、スライスレベルでのオーバーラップ実行による効果が得られる。ただし、設計者はユニット内での桁上げ信号のためのフィードバックループの存在に注意を払う必要がある。フィードバックループのようにデータの流と反対方向にデータが流れる箇所が存在すると、データの待ち合わせのために駆動パルス信号周波数の調整が必要となる。一方、ビットパラレル方式ではユニット内にフィードバックループは存在しない。そのため、3.3 節で説明したスキューを用いることで、駆動パルス信号周波数を演算ユニットに供給可能な最大値に設定することができる。

4.2 各アーキテクチャパラメータの評価目的と方法

本論文では SFQ 回路に適したマイクロアーキテクチャを検討すべく、各アーキテクチャパラメータが SFQ マイクロプロセッサの性能に及ぼす影響を明らかにするため性能モデルを用いて評価を行う。ここでは、1 命令あたりの実行に必要な時間 (TPI: Time Per Instruction) を用いる。プログラム実行時における TPI は式 (1) で求められる [5]。

$$TPI = \frac{T}{N_I} = \left(\frac{t_o}{\alpha} + \gamma \frac{N_H}{N_I} t_p \right) + \frac{t_p}{\alpha p} + \gamma \frac{N_H}{N_I} t_{op} \quad (1)$$

- T ：全命令の実行時間の総和。
- N_I ：全実行命令数。
- N_H ：発生するパイプラインハザードの総数。
- t_o ：パイプラインレジスタでのデータ保持に要する時間。ラッチのセットアップ/ホールドタイムにより決まる。
- t_p ：単一命令実行において通過する全論理ゲート遅延の総和。
- p ：パイプライン段数。

- α : スーパスカラ度.
- γ : パイプラインでの命令実行に要する時間 $t_{op} + t_p$ に対する, パイプラインハザードあたりの平均パイプラインストール時間の割合. 最大値は 1 であり, これは最初のパイプラインステージに存在する命令がコミットされるまで後続命令の実行が開始できない状況に相当する. 一方, 最小値は 0 であり, これはパイプラインストールが発生しない状況を表す. 実際には, 各種ハザードに起因するパイプラインストールはマイクロアーキテクチャとハザード発生状況によって様々であるが, 本モデルでは平均値として表す.

括弧でまとめた第 1 項はパイプライン段数に依存しない, ストールと t_o によって生じる実行時間の増加である. 第 2 項は 1 命令の処理に必要な時間がパイプライン段数とスーパスカラ度によってどの程度減少するのかを表している. 第 3 項はストールが発生することで生じる t_o の増加分を示す. ここで, 上記の TPI は次のように考えることができる. ストールがまったく起きない場合, すなわち $\gamma = 0$ のとき, TPI は 1 命令あたりの平均クロックサイクル数 (CPI: Clock Per Instruction) が 1 の場合, クロックサイクル時間 (逆数は動作周波数) と考えることができる.

評価ではスカラ・マイクロプロセッサ ($\alpha = 1$) を想定する. そして, ビットシリアル, ビットスライス, ビットパラレル方式を採用した SFQ マイクロプロセッサ (それぞれ, SFQ-BSE, SFQ-BSL, SFQ-BP と略す), ならびに, CMOS での設計を前提とした従来のビットパラレル型マイクロプロセッサ (CMOS-BP と略す) に着目した性能比較を行う. ここで, CMOS-BP はアウトオブオーダー命令発行方式を採用した商用プロセッサの構成に基づいて導入したモデルである. これは, 現在普及しているプロセッサ・アーキテクチャの一例であり, SFQ マイクロプロセッサのアーキテクチャ探索における比較基準として便宜上導入した. SFQ マイクロプロセッサと設計指針が類似する (すなわち, インオーダー命令発行方式を採用し, 高いクロック周波数の実現を目指す) CMOS マイクロプロセッサとの性能比較は 6 章で行う. SFQ マイクロプロセッサの遅延パラメータには, 現在多くの SFQ 回路で用いられている $1.0\ \mu\text{m}$ Nb プロセスの場合と, SFQ デバイスで成り立つスケールング則が限界に達する $0.3\ \mu\text{m}$ Nb プロセス [7] の 2 通りを用いる. t_o は SFQ マイクロプロセッサに用いられている代表的な論理ゲート群のセットアップ/ホールドタイムの算術平均と, 製造ばらつきやジッタ対策のための動作マージンを足し合わせて求めている. また, t_p は CORE 1 β のクリティカルパスである EX ステージを構成している SFQ 加算器の回路遅延に CORE 1 β のパイプライン段数である 7 を乗じて算出した. この SFQ 加算器の回路遅延は, 駆動パルス信号周期に SFQ 加算器で処理に要したサイクル数を乗じて求めることができる. 図 6 に SFQ 加算

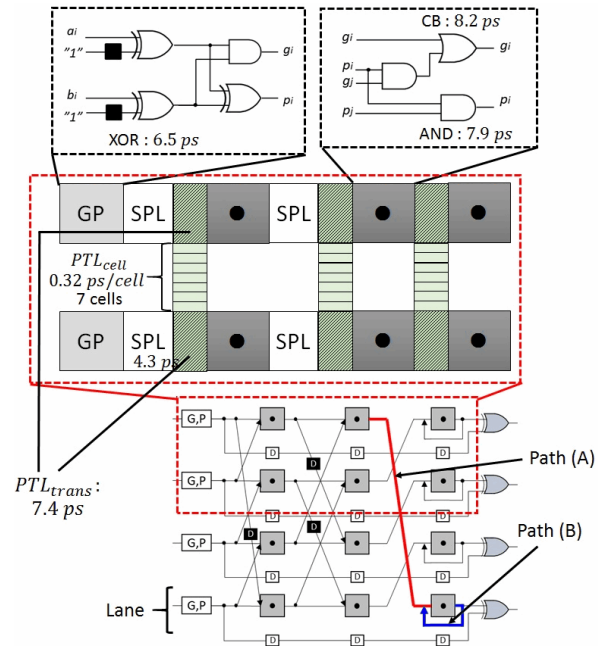


図 6 SFQ ビットスライス加算器のレイアウト

Fig. 6 Layout of an SFQ bit-slice adder.

器のレイアウトを示す. 回路遅延パラメータは $1.0\ \mu\text{m}$ Nb プロセスの SFQ セルライブラリの値を利用している. 以下で SFQ-BP と SFQ-BSE/BSL それぞれの場合の t_p の求め方について説明する.

SFQ-BP では, SFQ 加算器内にフィードバックループが存在しないため, 駆動パルス信号周波数は供給可能な最大値である t_o となる. また, 実際の設計ではワーストケースを想定し, $1.0\ \mu\text{m}$ Nb プロセスのとき $20\ \text{ps}$ としている.

SFQ-BSE/BSL では, SFQ 加算器内にフィードバックループが存在し, このループにおける入力データの待ち合わせが必要となる. よって, フィードバックループで計算されるデータ到着までの遅延がクリティカルパスとなり, 駆動パルス信号周期を決定する. 図 6 に示すように, ビットシリアル/スライス SFQ 加算器では, フィードバックループで計算されるデータが通るパスが 2 つある. これらをそれぞれパス (A), パス (B) とし, このどちらかがクリティカルパスとなる. この加算器では, パス (A) がパス (B) より長く, パス (A) に含まれる論理ゲート要素は, AND ゲート, コンフルエンスバッファ (CB: Confluence Buffer), 超伝導受動線路 (PTL: Passive Transmission Line), そしてスプリッタ (SPL: Splitter) である. PTL における信号伝搬時間は, 伝送線路に対する送受信に必要な遅延 PTL_{trans} と, 伝送線路の単位長さ (論理ゲートのセルサイズで規格化するものとする) あたりの伝搬遅延 PTL_{cell} の影響を受ける. 以上より, パス (A) の遅延 T_D は下記の式 (2) で表される.

$$T_D = AND + CB + PTL_{trans} + SPL + (dpw - 1) \times PTL_{cell} \times N_{PTLcell} \quad (2)$$

表 1 遅延パラメータ

Table 1 Delay parameters.

	t_o (ps)	t_p (ps)
1.0 μm SFQ-BP		2517.76
1.0 μm SFQ-BSE	13.32	13232.8
1.0 μm SFQ-BSL		4565.4
0.3 μm SFQ-BP		755.328
0.3 μm SFQ-BSE	3.995	3969.84
0.3 μm SFQ-BSL		1369.62
CMOS-BP	86.76	4048.58

ここで、AND、CB、SPLは、それぞれ、ANDゲート、コンフルエンスバッファ、スプリッタの遅延時間である。dpwはデータパスビット幅を指す。また、 $N_{PTLcell}$ は図6に示すような2つのビット線の間に存在するPTLセルの数である。ジョセフソン接合が $1/a$ にスケールすると、スイッチング速度と配線遅延も $1/a$ となる。求めたSFQ加算器の回路遅延はすべて64ビットのデータ語長の場合を想定している。SFQ-BSLに関しては、64ビットのデータ語長の場合に最も性能の良いスライス幅であった8ビットスライスのときの回路遅延を採用している。表1のビットシリアル/スライス方式の t_p が、ビットパラレル方式に対して、それぞれ単純に64倍、8倍の遅延時間となっていないのは、スライスデータのオーバーラップ実行が遅延時間の短縮に効いているためである。

CMOSマイクロプロセッサにおいては、Intel社の45nmプロセスのCore i7 920の構成を参考にし、パイプライン14段、動作周波数2.66 GHzを各パラメータ算出に用いた[6]。 t_o はパイプラインのラッチオーバーヘッドとパイプラインステージ遅延の比を最も良い比率とされている1.8対6と仮定し[5]、動作周波数から逆算して求めた $(1/2.66 [\text{GHz}] \times 1.8/(1.8+6))$ 。また、 t_p はCMOSマイクロプロセッサのクロック周期にパイプライン段数を乗じたものから、 t_o にパイプライン段数を乗じたものを引いた値 $(1/2.66 [\text{GHz}] \times p - t_o \times p)$ である。以上のパラメータについてまとめたものを表1に示す。

4.3 アーキテクチャパラメータの評価

図7にパイプライン段数を固定したとき(SFQは7段、CMOSは14段)の動作周波数を示す。まず、SFQ-BP/SFQ-BSE/SFQ-BSLの3つを比較する。1.0 μm Nbプロセスを想定した動作周波数(グローバルクロックの周波数)はそれぞれ2.78 GHz, 0.52 GHz, 1.53 GHz, 0.3 μm Nbプロセスの場合は9.27 GHz, 1.76 GHz, 5.11 GHzとなっている。ビットシリアル/スライス方式では、パイプラインステージ内の各論理ゲートが1語のデータ処理中に複数回活性化されるため t_p が増加し、その結果として動作周波数に大きな悪影響を及ぼす。このことから、ビットシリアル/スライス方式のオーバーラップ実行による時間短縮より、ビッ

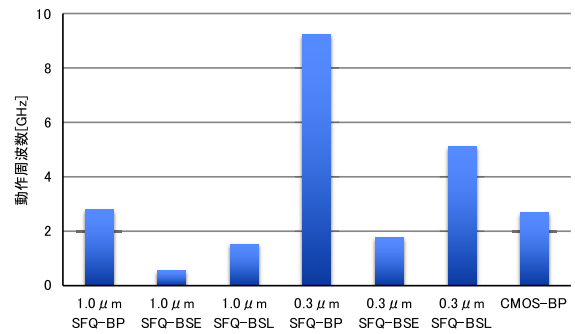


図 7 パイプライン段数を固定したときの動作周波数

Fig. 7 Clock frequencies with fixed pipeline stages.

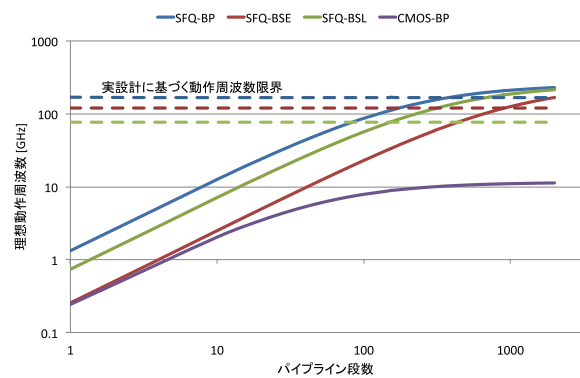


図 8 パイプライン段数を変化させたときの動作周波数

Fig. 8 Clock frequencies with variable pipeline stages.

トパラレル方式のビットレベル並列性の方が高性能化に効果的であるといえる。次に、CMOS-BPとの比較を行う。1.0 μm NbプロセスSFQマイクロプロセッサにおいて、SFQ-BP/SFQ-BSE/SFQ-BSLの動作周波数はCMOS-BPと同程度、もしくはそれ以下であることが分かる。一方、0.3 μm Nbプロセスにおいて、SFQ-BSEは1.76 GHzとCMOS-BPよりも低い動作周波数となっているものの、SFQ-BSLでは5.11 GHz, SFQ-BPでは9.27 GHzを達成可能である。SFQマイクロプロセッサが現状のCMOSマイクロプロセッサを凌駕する性能を達成するためには、ビットパラレル方式が最も適しているといえる。

図8に0.3 μm NbプロセスのSFQ-BP/SFQ-BSE/SFQ-BSLとCMOS-BPの動作周波数を示す。縦軸は理想動作周波数、横軸はパイプライン段数を表している。表1より、SFQ回路はCMOS回路と比較して t_o が非常に小さいといった利点を有することが分かる。そのためSFQマイクロプロセッサではパイプライン段数を深くすることで期待できる性能向上がCMOSに比べ大きい。また、CMOSマイクロプロセッサではパイプラインを深くすることによる動作周波数の向上は消費電力問題によりきわめて難しい。これに対し、SFQ回路ではその超低消費電力性から消費電力問題は生じない。そこで、パイプライン段数をより深くした場合を考える。図8における破線は、SFQ-BP/SFQ-BSE/SFQ-BSLの実設計データに基づく駆動パルス信号周

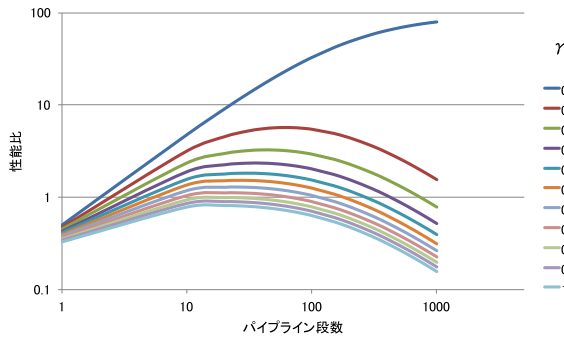


図 9 SFQ-BP と CMOS-BP の性能比較

Fig. 9 Performance comparison between bit-parallel SFQ microprocessor and CMOS microprocessor.

波数であり、それぞれ 166.67 GHz, 76.66 GHz, 119.90 GHz である。SFQ-BP のパイプライン段数を増加させた場合、377 段のときに SFQ-BP の動作周波数は 166.67 GHz に達成し、同パイプライン段数の SFQ-BSE/SFQ-BSL と比較しても高い性能である。一方、パイプライン段数を増加させた CMOS-BP は 11 GHz 程度である。以上のことから、SFQ マイクロプロセッサで超高速な動作周波数を得るためには、ビットパラレル方式かつ深いパイプライン構成をとる必要があると考える。ただし、式 (1) から分かるように、パイプライン段数を増加させると同時にパイプラインストールによる影響が大きくなり、TPI が増加する。

そこで、パイプラインストールが発生した場合の性能について考える。図 9 に 0.3 μm Nb プロセス、 $\frac{N_H}{N_I} = 0.5$ とした場合の SFQ-BP の γ の値を 0 から 1 まで変化させた場合の TPI の逆数を、CMOS-BP の動作周波数で正規化した結果を示す。横軸はパイプライン段数、縦軸は正規化された TPI の逆数であり、 $\gamma = 0$ 、すなわちストールが発生しない場合、SFQ-BP は現行 CMOS マイクロプロセッサに比べ 62.66 倍の性能となる。しかしながら、ストールの影響を反映した $\gamma = 0.1$ の場合でもパイプライン段数 60 段のときに最高 5.67 倍の性能しか達成できていない。つまり、パイプラインストールを十分に回避できなければ、高い動作周波数を達成できたとしても性能は低いものとなる。そこで、パイプラインストールを隠蔽できる場合について考える。新しくストール隠蔽率 θ ($0 \sim 1$, $\theta = 1$ のとき、ストールがすべて隠蔽されていることを示す) を導入し、ストール時間を表す γ と乗じる。 θ を導入した TPI のモデル式 (3) は以下のように表される。

$$TPI = \frac{T}{N_I} = \left(\frac{t_o}{\alpha} + (\gamma \times (1 - \theta)) \frac{N_H}{N_I} t_p \right) + \frac{t_p}{\alpha p} + (\gamma \times (1 - \theta)) \frac{N_H}{N_I} t_{op} \quad (3)$$

式 (3) に基づいて算出した $\frac{N_H}{N_I} = 0.5$, $\gamma = 0.5$ の場合の SFQ-BP の TPI の逆数を CMOS-BP の動作周波数で正規化した結果を図 10 に示す。横軸はパイプライン段数、縦軸は正規化された TPI の逆数である。この図 10 では、 θ

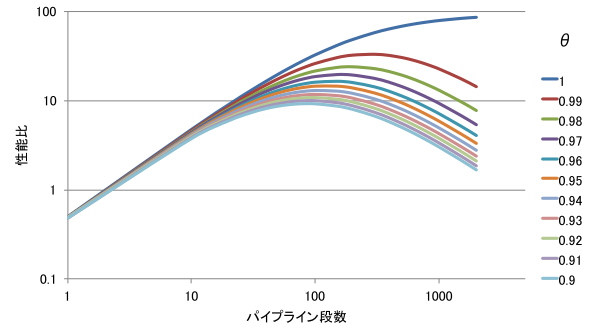


図 10 θ を考慮した SFQ-BP と CMOS-BP の性能比較

Fig. 10 Performance comparison between bit-parallel SFQ microprocessor and CMOS microprocessor considering θ .

の値を 0.9 から 1 まで 0.01 ずつ、すなわちパイプラインストール隠蔽率を 90% から 100% まで 1% ずつ増加させた場合の結果を示している。パイプラインストールを 99% 隠蔽できた場合、CMOS マイクロプロセッサに対する性能はパイプライン段数 300 段のときの 32.98 倍が最高となっている。このことから、パイプラインストールを 99% 隠蔽することでパイプライン段数が 300 段となるような非常に深いパイプライン構成をとったとしても性能向上が期待できることが分かる。また、高度なパイプラインストール隠蔽技術が必要となる。

5. SFQ マイクロプロセッサの論理設計

5.1 アーキテクチャの設計方針

SFQ マイクロプロセッサによる飛躍的な性能向上を実現するためには、デバイス/回路特性と各種設計制約を勘案したマイクロアーキテクチャを導入しなければならない。そこで我々は、4 章で示した性能評価結果、ならびに、SFQ 回路と設計技術に関する現状を考慮し、マイクロアーキテクチャ決定に向け以下の方針をとることとした。

5.1.1 ビットパラレル方式の採用

時間方向に処理を展開するビットシリアル/スライス方式と比較して、ビットパラレル方式は次に示す 3 つの利点を有する。1) 3.3 節で示したように、組合せ回路内部でのフィードバックループの発生を回避し、スキューの挿入により駆動パルス信号周波数を高めることができる。処理完了までのレイテンシは増大するが、これは後述する細粒度パイプライン処理でのオーバーラップ実行により隠蔽可能である。2) ビットレベル並列性を最大限に活用できるため、組合せ回路のレイテンシを低減させることができる。3) 時間方向の繰り返し処理が不要なため、回路設計ならびにレイアウト設計においてタイミング調整が比較的容易になる可能性がある。

5.1.2 ゲートレベル・パイプライン構成の採用

4.3 節で示したように、マイクロプロセッサレベル (すなわち、駆動パルス信号ではなくパイプライン全体の動作速度を決定するグローバルクロック信号レベル) での高速動

作を可能にするためには、きわめて深いパイプライン構成をとる必要がある。そこで、各パイプラインステージが1段の論理ゲートで構成されるゲートレベル・パイプライン構成をとる。これにより、従来のSFQマイクロプロセッサで必要であった駆動パルス信号とグローバルクロック信号を一本化（つまり、駆動パルス信号でパイプライン全体の動作を制御する）でき、設計容易化の観点からも利点がある。

5.1.3 大規模細粒度マルチスレッディング実行方式の採用

4.3節で示したように、細粒度のパイプライン構成においてCMOSマイクロプロセッサを大幅に超える性能を実現するためには、ハザードに起因するパイプラインストール時間を99%以上隠蔽する必要がある。既存の高性能マイクロプロセッサではストール隠蔽技術としてアウトオブオーダー実行方式を採用している。しかしながら、パルス論理に基づくSFQ回路ではタイミング調整がクリティカルな設計となり、複雑な論理を要する命令ウィンド、リネーミング機構、高度な分岐予測機構、リオーダーバッファ、パイプラインステージ間をまたいだフィードバックループを形成するフォワーディング機構などを実装することが難しいと予想される。そこで、徹底したハードウェアの簡素化を実現しつつ、きわめて深いパイプラインでのストール隠蔽を可能にすべく、パイプライン段数と同程度のスレッドを起動しクロックサイクルごとに切り替えて実行を進める大規模細粒度マルチスレッディング方式を導入する。

5.1.4 FIFOメモリを基本とするメモリ階層の採用

大規模細粒度マルチスレッディングを実現するためには、巨大なレジスタファイルを構成し、クロックサイクルごとに適切なレジスタセットを後段パイプラインステージへと出力しなければならない。SFQ回路を用いたメモリの実装に関しては、これまでにいくつかの提案が行われてきたが、最も実用的なのはシフトレジスタを基本とするFIFOメモリである。これは、SFQ回路はフィードバックループを持たない単方向データ流の処理を得意とすることに起因する。たとえば、 $2.0\mu\text{m}$ Nbプロセスにおいて20psで動作可能なシフトレジスタの設計事例があり[4]、SFQマイクロプロセッサCORE1 β のレジスタファイル[11]、[16]や、CORE1 γ [13]のキャッシュメモリとして実装されている。そこで、FIFOメモリを用いた大規模細粒度マルチスレッディング向けレジスタファイルを導入する。

5.2 設計目的

本設計の目的は、5.1節で述べた方針に基づきSFQマイクロプロセッサの性能、回路規模、消費電力を見積もり、実現可能性と有効性を評価することである。特に、論理ゲートレベルの深いパイプライン構成を採用した場合、その段数は最も長いデータパスに合わせる必要があり、タイミング調整のためのラッチ回路の挿入が必要となる（これをダミーラッチと呼ぶ）。このダミーラッチの多寡が回路規模

と消費電力に大きく影響する。この影響を明かにするためにはパイプライン段数（すなわち、最も長いデータパスを構成する論理ゲート段数）を明かにする必要がある。ハードウェア記述言語であるVerilog HDL、および、Cadence社のシミュレータNC-Verilogを用いて実際の論理設計を行った。

5.3 設計仕様

設計するマイクロプロセッサの命令長は32ビット、データ語長は8ビットである。アドレス空間は32ビットであり、データメモリへのアクセス時には32ビットに符号拡張される。本評価ではSFQマイクロプロセッサのポテンシャルを明らかにすることを目的とし、メモリアクセスは理想的な状態（つまり、1クロックサイクルでアクセスが完了する）ことを想定する。命令セットアーキテクチャに関してはプログラム実行に最低限必要となる基本命令のみに限定し、MIPS ISAで定義される命令から、加減算（*add*, *addi*, *sub*）、メモリ参照（*load*, *store*）、条件付き分岐（*beq*, *bne*）、ならびに、無条件ジャンプ（*j*, *jr*, *jal*）を選択した。図11に設計した8ビットSFQマイクロプロセッサのブロック図を示す。1命令の実行を完了するまでに通過する全論理ゲート段数は52段となった。なお、本設計において、*store*命令の実行では正しい動作が確認できていない。これは、各パイプラインレジスタの初期化を行っておらず、それにともない不定値が伝搬したことが原因である。SFQマイクロプロセッサによる初期化方法は検討中であり今後の課題となるが、本設計に基づく性能、回路規模、ならびに、消費電力に関する評価結果への影響は無視できる程度に小さいと考える。

5.4 64ビットへの拡張

本論文で設計した8ビットSFQマイクロプロセッサを64ビットCMOSマイクロプロセッサと比較するために、8ビットの設計結果から64ビットへ拡張した場合の所要パイプライン段数を見積もる。64ビットSFQマイクロプロセッサへ拡張するにあたり、設計した8ビットSFQマイクロプロセッサにおいてパイプライン段数が増加するデータパスは以下の3つである。

- (1) 8ビットALUが存在するデータパス
- (2) 後続命令のPC計算値、分岐先アドレス計算のための32ビットの加算器が存在するデータパス
- (3) 分岐判定のための入力値の一致判定回路（8ビット）

上記のうち、(1)および(2)については、ALUはSFQ Kogge-Stone adder[19]を用いた加算器を想定すると、パイプライン段数 P_{adder} はビット幅 x を用いて式(4)で表される。

$$P_{adder} = 1 + (\log_2 x) \times 2 + 2 \quad (4)$$

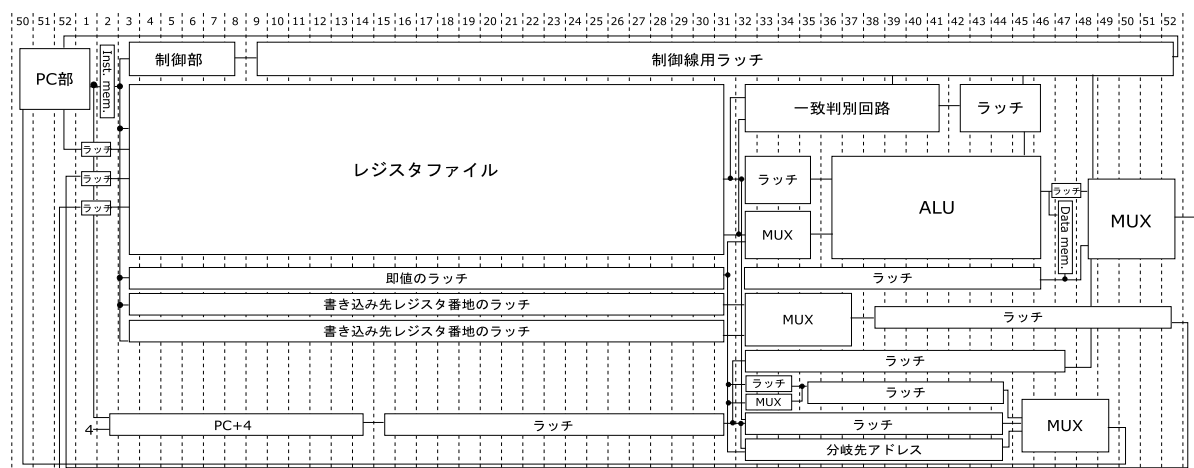


図 11 8ビット SFQ マイクロプロセッサのブロック図
Fig. 11 A block diagram of the 8-bit SFQ microprocessor.

ただし、ALU では減算を可能にするため、最初に加算か減算かを判別し、減算の場合は 1 の補数表現をとる回路が挿入されている。その回路によってパイプライン段数が 3 段多くなり、64 ビットの SFQ Kogge-Stone adder のパイプライン段数は 15 段、ALU 全体ではこれに 3 段加えて 18 段となる。また、分岐先アドレス計算は加算減算判別が必要ないので 13 段である。一方、分岐判定のための入力値の一致判定回路 (8 ビット) では、各ビットの一致判定に 3 段必要であり、全ビットの論理積をとるために 3 段、そして分岐命令が *beq* か *bne* によって出力する判定結果を選択するために 4 段の計 10 段のパイプライン構成となる。この回路が 64 ビットに拡張された場合、集約論理積演算のために必要なパイプライン段数 P_{AND} が増加し、式 (5) によって求められる。64 ビットの場合は 6 段、8 ビットの場合は 3 段であり、3 段の増加となる。

$$P_{AND} = \log_2 x \quad (5)$$

以上より、64 ビットに拡張した際に増加するパイプライン段数は (1), (2), (3) のデータパスでそれぞれ 6 段、2 段、3 段となり、上記 3 つの中で最もパイプライン段数が増加するデータパスは (1) の ALU が存在するデータパスで 6 段である。したがって 64 ビットに拡張した際のパイプライン段数は、8 ビットのパイプライン段数 52 段に 6 段加えた 58 段となる。

6. 評価

6.1 性能評価

本評価ではまず、設計した 8 ビット SFQ マイクロプロセッサに基づき、動作周波数を算出する。設計した SFQ マイクロプロセッサには演算結果をレジスタファイルへ転送する部分を除きフィードバック回路が存在しないため、3.3 節で述べたスキュー挿入を前提とする。駆動パルス信号線にスキューを挿入することで、ゲートを通過する際の

表 2 SFQ 論理ゲートのセットアップタイム、ホールドタイム

Table 2 Setuptime and holdtime of SFQ logic gates.

論理ゲート名	セットアップタイム	ホールドタイム
ラッチ	0.36(ps)	-0.27(ps)
AND	-0.54(ps)	0.81(ps)
NOT	2.22(ps)	0.06(ps)
OR	1.74(ps)	-0.06(ps)
XOR	1.11(ps)	1.23(ps)

データ信号の遅延が隠蔽可能となる。この場合、クロックサイクルタイム CCT は以下の式 (6) で求められる。

$$CCT = (HoldTime_{N-Gate} + M) + (SetupTime_{N-Gate} + M) \quad (6)$$

ここで $HoldTime_{N-Gate}$ は次段論理ゲートのホールドタイム、 $SetupTime_{N-Gate}$ は次段論理ゲートのセットアップタイム、 M は製造ばらつきやジッタ対策のためのマージンを表している。本評価で用いた各 SFQ 回路要素のセットアップタイム、ホールドタイムを表 2 に示す。セットアップタイム、ホールドタイム、マージンの値はジョセフソン接合が $1/\alpha$ にスケールするとすべて $1/\alpha$ となる。表 2 の値はすべて $1.0 \mu\text{m Nb}$ プロセスの値を基に $0.3 \mu\text{m Nb}$ プロセスの値を算出したものである。表 2 にマイナスの値が存在するが、それは駆動パルス信号とデータパルス信号の入力されるタイミングが前後するという意味である。たとえば、セットアップタイムの場合、3.3 節の定義の逆となるため、データパルス信号入力前にどれだけ早く次の駆動パルスを入力することが可能かという時間である。これは論理ゲートの構成に起因する。マージンの値も同様に $1.0 \mu\text{m Nb}$ プロセスの場合の $M = 3.6 \text{ (ps)}$ を基に $0.3 \mu\text{m Nb}$ プロセスの場合 $M = 1.08 \text{ (ps)}$ と算出した。ここで、クロックマージンに関しては、本来であればレイアウト設計に基づく遅延情報抽出を行い、タイミング制約を満足する適切な

表 3 比較対象の CMOS マイクロプロセッサ・モデル
Table 3 The CMOS microprocessor models to compare.

比較対象	動作周波数	パイプライン段数	スーパースカラ度	消費電力
Cell SPE-HP	5.0 GHz	26 段	2	11 W
Cell SPE-LP	3.2 GHz	26 段	2	3 W

値を設定する必要がある。しかしながら、現段階では論理設計までしか実施できておらず、過去の設計事例を参考にクロックマージンの値を設定した。しかしながら、これまでに開発された SFQ マイクロプロセッサと比較して本論文で提案するアーキテクチャでは回路が大規模化するため、より大きなクロックマージンが必要となる可能性がある。レイアウト設計後のタイミング検証に基づく、より正確なクロックマージンの設定と詳細な評価については今後の課題である。本評価では、セットアップ・ホールドタイムとマージンの和が最も大きくなった XOR ゲートの値をクロックサイクルタイムとする。したがって、設計した SFQ マイクロプロセッサの動作周波数は 222.22 GHz である。この結果は 4.3 節で求めた最大動作周波数 166.67 GHz より大きな値となっているが、これはスキュー挿入によって論理ゲート遅延の隠蔽効果が現れたためである*2。

次に、CMOS マイクロプロセッサと 64 ビット SFQ マイクロプロセッサの比較を行う。異なるマイクロプロセッサの性能を比較する場合、動作周波数だけでなく、各種依存関係やメモリアクセスによって生じるストールの発生、同時に実行可能な命令数など様々な影響を反映する必要がある。また、使用するデバイス（プロセステクノロジーなど）やチップ面積も考慮しなければならない。しかしながら、一般的にこれらすべてを考慮した公平な評価はきわめて困難である。また現段階では SFQ 回路の自動設計技術が未熟なため、詳細なレイアウト設計まで考慮した性能評価ができない。そこで本論文では、第ゼロ次近似評価として、SFQ マイクロプロセッサと同じ設計指針に基づく商用 CMOS マイクロプロセッサの構成を参考にした性能評価を実施する。なお、SFQ マイクロプロセッサにおける冷却オーバーヘッドやレイアウト設計の影響などを考慮した厳密な評価の実施は今後の課題である。

近年のマイクロプロセッサは、消費電力削減を重視するため動作周波数を低く設定する傾向にある。これに対し、SFQ マイクロプロセッサは、きわめて単純な構成を採用し、動作周波数を最大に高めるアプローチをとっている。そこで本評価では、これらの 2 つの設計指針を基本にしていると考えられる最近の商用マイクロプロセッサとして Cell Broadband Engine Synergistic Processor Element (90nm) [2], [3] に着目し、シングル・コアのマイクロアー

キテクチャを参考としてハイパフォーマンス時、および省電力動作時の 2 つを評価対象モデルに設定した*3。設定したそれぞれのモデルを Cell SPE-HP モデル、Cell SPE-LP モデルとし、それらの詳細を表 3 に示す*4。評価指標として TPI、および単位時間あたりの平均命令実行数（BIPS: Billion Instructions Per Second）を用いる。BIPS は動作周波数（GHz）と 1 クロックサイクルあたりの命令実行数（IPC: Instructions Per Clock）の積で求められる。ここで、IPC は、スーパースカラ度 α とパイプラインストール率 ω 、ならびに、パイプライン段数 p を用いて式 (7) で求めることができる。 ω はパイプラインステージの平均ストール率であり、1 の場合は $p-1$ 段のパイプラインがストールしている状態（1 命令のみパイプライン上に存在する）を表す。

$$IPC = \alpha / (1 + \omega \times (p - 1)) \quad (7)$$

TPI はスキューの影響を明らかにするため、また、BIPS はマルチスレッディングでストールが隠蔽できない（対象のアプリケーションによって細粒度マルチスレッディングが実行できない）場合のストールの影響を明らかにするために評価指標として用いた。スキューの影響を明らかにするためには、挿入されたスキューによる遅延の和を求める必要があるが、これはレイアウト設計を行わなければ求めることができない。そこで、スキューが実行パスの全ステージに一樣に挿入されると仮定する。1 パイプラインステージあたりの遅延、すなわちクロックサイクルタイム+1 パイプラインステージあたりのスキューの、1 クロックサイクルタイムに対する比率を表すスキュー率 τ を導入する。 τ はクロックサイクルタイム CCT と 1 パイプラインステージあたりのスキューによる遅延 D_{skew} を用いて以下の式 (8) で表すことができる。

$$\tau = (CCT + D_{skew}) / CCT \quad (8)$$

これにより、スキュー率とパイプライン段数の積からプロセッサに挿入されるスキューによる遅延の和を算出し、スキュー挿入による影響を明らかにすることができる。図 12

*2 4.3 節ではマイクロアーキテクチャ・レベルでの比較検討を行うことを目的とし、論理ゲートレベルの内部構造に強く依存するスキュー挿入は前提としなかったが、議論の一般性は失われない。

*3 この商用マイクロプロセッサの世代を最後に、その後のアーキテクチャ・トレンドは消費電力の増加を抑制するため動作周波数を低く抑え、かつ、プロセッサコア数を増大させることでスループットを向上させる方向へとシフトした。

*4 本評価の結果は、本論文で定めた性能モデルに基づき、仮想的に設定した評価対象モデルと比較したものであり、商用の Cell B.E. との厳密な比較ではないことに注意されたい。

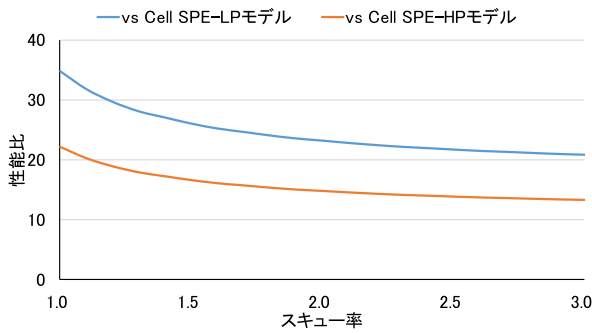


図 12 TPI における性能比較

Fig. 12 TPI performance comparison between SFQ and CMOS microprocessors.

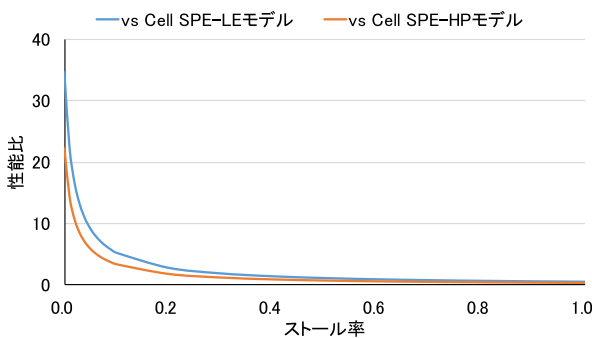


図 13 BIPS における性能比較

Fig. 13 BIPS performance comparison between SFQ and CMOS microprocessors.

に CMOS マイクロプロセッサ・モデルに対する SFQ マイクロプロセッサの TPI 性能比を示す。図 12 の縦軸は、Cell SPE-HP モデルならびに Cell SPE-LP モデルそれぞれに対する SFQ マイクロプロセッサの性能比 (TPI の逆数の比) を示している。ここで、CMOS での実装を前提としている Cell SPE-HP ならびに Cell SPE-LP モデルでは理想的な場合を想定し、パイプラインストール率 $\omega = 0$ 、IPC=スーパースカラ度 (すなわち 2.0) を前提とした。横軸は SFQ マイクロプロセッサの設計におけるスキュー率を表しており、本評価では 1.0 から 3.0 まで変更した場合を想定した。これまでの設計実績によると、スキュー率は 2.0 程度になることが経験的に分かっている [19]。スキュー率が 2.0 の場合、Cell SPE-HP モデルに対しては約 15 倍、Cell SPE-LP モデルに対しては約 23 倍の性能を達成していることが分かる。また、CMOS マイクロプロセッサ・モデルに対する SFQ マイクロプロセッサの BIPS 性能比を図 13 に示す。ここで、横軸は仮定となる SFQ マイクロプロセッサのパイプラインストール率を表す。その他の前提や図の見方は図 12 と同様である。ストールが発生しない場合、Cell SPE-HP モデルに対して約 22 倍、Cell SPE-LP に対して約 35 倍の性能を達成しているが、ストール率 $\omega = 0.1$ 、すなわちストールが 10%発生した場合、それぞれ約 5 倍、約 3 倍の性能向上しか得られないことが分

表 4 パラメータ

Table 4 Parameters.

パラメータ	値
I_c	0.1(mA)
V	2.5(mV)
I_{bias}	150(μ A)

表 5 SFQ 論理ゲートの JJ 数

Table 5 The number of JJs of SFQ logic gates.

論理ゲート名	JJ 数
ラッチ	6
AND	15
NOT	11
OR	12
XOR	11

かる。これは、非常に深いパイプライン構成をとっており、ストールの影響が大きいと思われる。そのため、設計した SFQ マイクロプロセッサで高い性能を達成するためには、細粒度のマルチスレッディングによりほとんどのストールを隠蔽する必要がある。

本評価結果より、スーパースカラ度が 2 である CMOS マイクロプロセッサに対し、単一命令発行を前提とした SFQ マイクロプロセッサにおいても大幅な性能向上が期待できることが明らかになった。フィードバックループを有する複雑なランダムロジックに関して、SFQ 回路でターゲットとしている数十 GHz 以上の動作周波数では電磁波の信号伝搬時間が無視できないためその実現は難しい。また、量子を情報担体とするためファンアウトが得られにくく、メモリにおける入出力ポートの増加は容易ではない。以上より SFQ 回路は、スーパースカラ度を低く抑えたインオーダー命令発行型プロセッサの実現に適していると考えられる。

6.2 消費電力

本節ではジョセフソン接合数に基づき消費電力の評価を行う。SFQ マイクロプロセッサの全体消費電力 P_{all} は動的消費電力 $P_{dynamic}$ と静的消費電力 P_{static} を用いて式 (9) で表される。

$$P_{all} = P_{dynamic} + P_{static} \quad (9)$$

また、 $P_{dynamic}$ 、 P_{static} はそれぞれ式 (10)、式 (11) で与えられる。

$$P_{dynamic} = \alpha \Phi_0 I_c f \times N_{JJ} \quad (10)$$

$$P_{static} = V I_{bias} \times N_{JJ} \quad (11)$$

ここで、 α はスイッチング確率、 Φ_0 は磁束量子、 I_c は臨界電流値、 f は動作周波数、 V はバイアス電流の制御用に用いられる電源電圧、 I_{bias} は 1 ジョセフソン接合あたりに流れるバイアス電流、 N_{JJ} はジョセフソン接合数である。 I_c 、 V 、 I_{bias} の値は過去の設計事例 [17] に基づき、表 4 に示す値を用いている。

SFQ マイクロプロセッサの全ジョセフソン接合数 JJ_{all} は、論理ゲートを構成するジョセフソン接合数と配線部分のジョセフソン接合数を用いて式 (12) で表される。

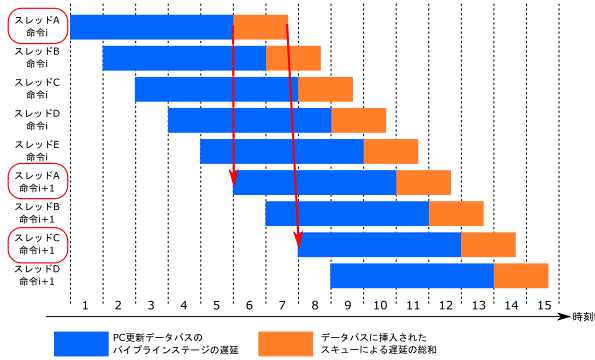


図 14 スキュー挿入による PC 読み込み時刻のずれ

Fig. 14 Reading different program counter caused by inserting skew elements.

$$JJ_{all} = JJ_{logic} + JJ_{wire} \quad (12)$$

JJ_{logic} は各 SFQ 論理ゲートを構成するジョセフソン接合数 (表 5 [17]), 設計した SFQ マイクロプロセッサの回路図, およびダミーラッチ数から求めることができる. 一方, JJ_{wire} はレイアウト設計を行わなければ求めることができない. 本評価では, レイアウト設計は実施していないため, 論理ゲート部分の 1 つのジョセフソン接合に対する配線部分のジョセフソン接合数を κ とおき, 式 (13) で求める. κ は過去の設計事例 [12] より 0.798 として評価を行う.

$$JJ_{wire} = JJ_{logic} \times \kappa \quad (13)$$

次に, スキューによる起動スレッド数の増加にともなうシフトレジスタ面積の増加について考える. 3.3 節で述べたスキュー挿入によって動作周波数を最大値に設定可能だが, 細粒度マルチスレディングの動作に問題が生じる. スキューの遅延によって PC の更新が間に合わず, 本来同一スレッドの先行命令の PC の更新結果を正しく読み出すことができないという問題である. その様子を図 14 に示す. この図 14 ではスレッド A/B/C/D/E が巡回実行される場合を想定している. 本来スレッド A の命令 i は第 5 クロックサイクルで終了するため, 第 6 クロックサイクル目から再びスレッド A の実行が可能であるが, スキュー挿入によりスレッド A の命令 i の実行が 7 クロックサイクル必要となってしまう. その結果, スレッド A の命令 i の終了前 (PC 更新前) に次の命令がフェッチされてしまい, 命令を正しく読み出すことができない. これを回避するため, 下記の式 (14) で求められる数のスレッドを追加で起動する必要がある.

$$T_{num} = \text{ceil}(skew_{all}/CCT) \quad (14)$$

CCT はクロックサイクルタイム, $skew_{all}$ は挿入されたスキューすべてによる遅延時間の和である. $skew_{all}$ は 6.1 節で導入したスキュー率とパイプライン段数の積から算出した.

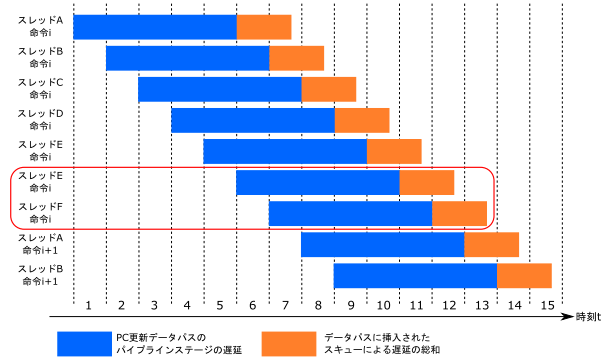


図 15 起動スレッドの追加によるタイミング調整

Fig. 15 Timing adjustment by adding threads.

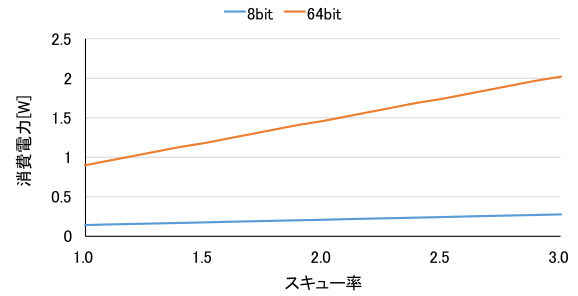
図 16 $\alpha = 1$ の場合の SFQ マイクロプロセッサの消費電力

Fig. 16 Power consumption of the SFQ microprocessors.

図 14 の場合は追加起動するスレッド数 T_{num} は 2 となり, その際の実行の様子を図 15 に示す. この場合, パイプライン段数以上のスレッドが起動することになるが, SFQ 回路のパルス論理では配線上に複数パルスが存在可能であるため, スキューによって各パイプラインステージにおける単一配線上に複数命令が存在しても問題はない. また, 起動スレッド数に合わせてレジスタファイルのシフトレジスタの深さを増加させる. これは, 各スレッドのアーキテクチャステートのレジスタセットを保持できるようにするためである. PC 部にも同様のシフトレジスタを実装し, 起動スレッド数に合わせてシフトレジスタの深さを調整する.

図 16 に 8 ビット, 64 ビット SFQ マイクロプロセッサの消費電力を示す. 8 ビット, 64 ビット SFQ マイクロプロセッサの消費電力は, スキュー率 τ が 2, ワorst ケースを想定したスイッチング確率 α が 1 の場合にそれぞれ 0.207 W, 1.462 W であることが分かる. 表 3 より Cell SPE-HP および Cell SPE-LP モデルの消費電力はそれぞれ 11 W, 3 W であるが, この値は Cell Broadband Engine Synergistic Processor Element において単精度浮動小数点演算インテンシブなアプリケーションをそれぞれ 5.0 GHz, 3.2 GHz で実行した際に消費される電力値に基づいている [3]. 設計した SFQ マイクロプロセッサは単精度演算器を搭載していないため, Cell SPE-HP, Cell SPE-LP モデルとの正当な比較はできない. そこで過去に SFQ 回路を

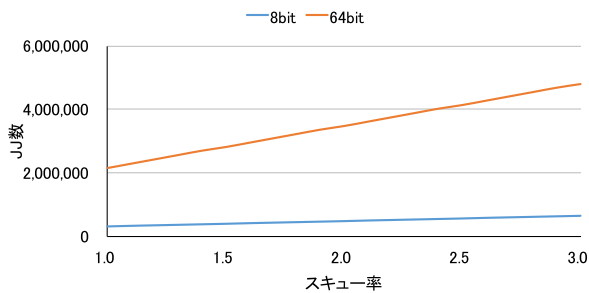


図 17 SFQ マイクロプロセッサのジョセフソン接合数

Fig. 17 The number of JJs of the SFQ microprocessors.

用いて設計されたビットシリアル型単精度浮動小数点演算器（単精度加算器と単精度乗算器）の消費電力を加味した比較を行う [10]. ビットシリアル型 SFQ 単精度浮動小数点加算器と乗算器の消費電力の和は, SFQ 回路の低消費電力化技術である LR バイアス [15] の利用を想定した場合 0.51 mW であることが報告されており, この値を参考にして 32 ビットパラレル方式に拡張する場合の電力を見積もる. SFQ 回路の消費電力は前述したとおり回路規模に大きく依存する. 文献 [10] で設計された浮動小数点演算器の構成はシストリックアレイタイプであり, 実際には通常のビットシリアル設計より大規模な構成となっている. したがって回路規模が線形増加するとは考えにくい, 本評価では SFQ にとって悲観的な観点での比較を行うため 32 ビットパラレル拡張した浮動小数点演算器の電力をビットシリアル型浮動小数点演算器の 32 倍として見積もる. よって, 32 ビットパラレルに拡張した場合の消費電力は 16.32 mW であり, 設計した SFQ マイクロプロセッサに単精度浮動小数点積和演算器を搭載した場合の消費電力は約 1.48 W と見積もられる. これは Cell SPE-HP モデルの消費電力の約 1/7 倍, Cell SPE-LP モデルの約 1/2 倍である. このことからゲートレベル命令パイプラインを採用した SFQ マイクロプロセッサは 222.22 GHz という超高動作周波数にもかかわらず, 非常に低い消費電力で動作することの可能性を確認できる. また, 様々な SFQ 低消費電力技術が新たに考案されており [8], [14], [15], 実際には SFQ マイクロプロセッサのさらなる低消費電力化が期待できる.

6.3 面積

本評価では 6.2 節の消費電力評価と同様にジョセフソン接合数を基準に面積を見積もる. 図 17 に $\kappa = 0.798$ の場合の 8 ビット, 64 ビットの SFQ マイクロプロセッサのジョセフソン接合数を示す. 横軸はスキュー率であり, 縦軸はジョセフソン接合数である. 上記の消費エネルギー評価と同様 κ は 0.798 とした. 図 17 より, 8 ビットではスキュー率 τ が 2 の場合, ジョセフソン接合数は 470,971 であることが分かる. また, 64 ビットへと拡張した場合には,

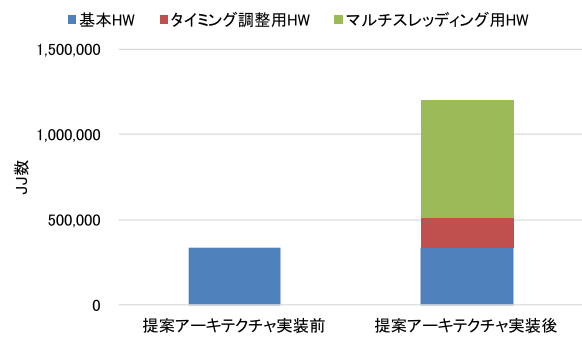


図 18 配線部分を除く 64 ビット SFQ マイクロプロセッサの面積内訳

Fig. 18 Area breakdown of the 64-bit SFQ microprocessor.

スキュー率 τ が 2 のときジョセフソン接合数は 3,473,306 である. 8 ビット SFQ マイクロプロセッサと比べて 7.33 倍と面積が増加している. この面積の増加の原因を明かにすべく, ゲートレベル命令パイプラインと細粒度マルチスレディングを採用したことによる面積の増加を解析する. 図 18 に配線部分を除く 64 ビット SFQ マイクロプロセッサの面積の内訳を示す. 図 18 中の「基本 HW」はゲートレベル命令パイプラインをとるが細粒度マルチスレディングは未実装の場合である. また, 「タイミング調整 HW」はゲートレベル命令パイプラインを採用したことによって増加したタイミング調整用の回路面積, 「マルチスレディング HW」は細粒度マルチスレディングを追加したことによって追加されたレジスタファイル部などの面積である. 図 18 より, 提案アーキテクチャ実装によって実装前と比較し面積が 3.58 倍になることが明らかになった. また, 基本 HW は全体の約 1/4 であるのに対し, タイミング調整 HW とマルチスレディング HW が残りの約 3/4 を占めていることが分かる. 特に, マルチスレディング実装のために追加されたシフトレジスタの面積が過半数を占め, 細粒度マルチスレディングによる面積のオーバーヘッドは非常に大きい. これより, 細粒度マルチスレディングを実装するために必要な複数のアーキテクチャステートを持するレジスタが占める面積が大きいことが分かる. 面積を削減するために, 今後スキューの挿入する条件を改善する必要がある. それによりスキュー率が低下し, 結果として面積を削減可能であると考えられる.

7. おわりに

本論文では, 超高性能マイクロプロセッサの実現に向け, SFQ 回路を利用したマイクロプロセッサのアーキテクチャ設計方針を検討し, 期待できる性能向上やオーバーヘッドを明らかにするためにゲートレベルで理論設計し評価した. 評価の結果, 論理ゲートレベルの命令パイプラインを構成することで 222.22 GHz の動作周波数を達成可能であることを考察した. また, パイプライン段数の増加にともなうストール時のペナルティを回避するため, 細粒度マルチス

レッディングを採用することでデータ依存および制御依存によるストールの発生を抑止可能であることが分かった。

SFQ回路を用いたマイクロプロセッサのアーキテクチャ設計方針に基づき、SFQマイクロプロセッサの論理設計を行い性能、消費電力、および、面積を算出した。その結果、設計指針が類似する商用CMOSマイクロプロセッサの構成を参考としたモデルと比較して最大約35倍の性能を達成可能であることが分かった。また、1.46 Wと低消費電力で動作する可能性があり、その際の実面積は提案アーキテクチャ実装前と比較して3.58倍になることが明らかになった。なかでも細粒度マルチスレッディングによって追加されるレジスタファイルがプロセッサ全体の約半分を占めることが分かった。

SFQ回路の面積の削減手法の検討やSFQ回路とCMOS回路の面積を定量的な比較、およびアプリケーションを用いた実性能評価は今後の課題である。本論文ではプロセッサのアーキテクチャに着目したが、SFQプロセッサにデータを供給するために必要なメモリやキャッシュメモリの検討を実施する予定である。また、SFQは冷却設備などほかにも電力を必要とする装置が存在することから、プロセッサだけでなくこれらを含めた計算機システムの評価が必要である。

謝辞 本研究を進めるにあたり、活発な議論とご協力をいただいた九州大学井上研究室の皆様にご心より感謝の意を表します。なお、本研究は、一部文部科学省科学研究費補助金16H02796ならびにALCA-JSTの助成による。

参考文献

- [1] Anderson, P.W. and Rowell, J.M.: Probable Observation of the Josephson Superconducting Tunneling Effect, *Phys. Rev. Lett.*, Vol.10, pp.230–232 (online), DOI: 10.1103/PhysRevLett.10.230 (1963).
- [2] Chen, T., Raghavan, R., Dale, J.N. and Iwata, E.: Cell Broadband Engine Architecture and Its First Implementation: A Performance View, *IBM J. Res. Dev.*, Vol.51, No.5, pp.559–572 (online), DOI: 10.1147/rd.515.0559 (2007).
- [3] Flachs, B., Asano, S., Dhong, S.H., Hofstee, H.P., Gervais, G., Kim, R., Le, T., Liu, P., Leenstra, J., Liberty, J., Michael, B., Oh, H.-J., Mueller, S.M., Takahashi, O., Hatakeyama, A., Watanabe, Y., Yano, N., Brokenshire, D.A., Peyravian, M., To, V. and Iwata, E.: The microarchitecture of the synergistic processor for a cell processor, *IEEE Journal of Solid-State Circuits*, Vol.41, No.1, pp.63–70 (online), DOI: 10.1109/JSSC.2005.859332 (2006).
- [4] Fujiwara, K., Yamashiro, Y., Yoshikawa, N., Fujimaki, A., Terai, H. and Yorozu, S.: Design and high-speed test of (4×8) -bit single-flux-quantum shift register files, *Superconductor Science and Technology*, Vol.16, No.12, p.1456 (2003).
- [5] Hartstein, A. and Puzak, T.R.: The optimum pipeline depth for a microprocessor, *ACM SIGARCH Computer Architecture News*, Vol.30, No.2, pp.7–13, IEEE Computer Society (2002).
- [6] Hennessy, J.L. and Patterson, D.A.: *Computer Architecture, Fifth Edition: A Quantitative Approach, 5th edition*, Morgan Kaufmann Publishers Inc. (2011).
- [7] Kadin, A.M., Mancini, C.A., Feldman, M.J. and Brock, D.K.: Can RSFQ logic circuits be scaled to deep sub-micron junctions?, *Applied Superconductivity, IEEE Transactions on*, Vol.11, No.1, pp.1050–1055 (2001).
- [8] Kirichenko, D.E., Sarwana, S. and Kirichenko, A.F.: Zero Static Power Dissipation Biasing of RSFQ Circuits, *IEEE Trans. Applied Superconductivity*, Vol.21, No.3, pp.776–779 (online), DOI: 10.1109/TASC.2010.2098432 (2011).
- [9] Likharev, K.K. and Semenov, V.K.: RSFQ logic/memory family: A new Josephson-junction technology for sub-terahertz-clock-frequency digital systems, *IEEE Trans. Applied Superconductivity*, Vol.1, No.1, pp.3–28 (online), DOI: 10.1109/77.80745 (1991).
- [10] Peng, X., Xu, Q., Kato, T., Yamanashi, Y., Yoshikawa, N., Fujimaki, A., Takagi, N., Takagi, K. and Hidaka, M.: High-Speed Demonstration of Bit-Serial Floating-Point Adders and Multipliers Using Single-Flux-Quantum Circuits, *IEEE Transactions on Applied Superconductivity*, Vol.25, No.3, pp.1–6 (online), DOI: 10.1109/TASC.2014.2382973 (2015).
- [11] Tanaka, M., Kawamoto, T., Yamanashi, Y., Kamiya, Y., Akimoto, A., Fujiwara, K., Fujimaki, A., Yoshikawa, N., Terai, H. and Yorozu, S.: Design of a pipelined 8-bit-serial single-flux-quantum microprocessor with multiple ALUs, *Superconductor Science and Technology*, Vol.19, No.5, p.S344 (2006).
- [12] Tanaka, M., Takata, K., Satoh, R., Fujimaki, A., Kawaguchi, T., Ando, Y., Takagi, K., Takagi, N. and Yoshikawa, N.: Design of RSFQ Microprocessors Integrated with RAMs Based on Bit-Serial Processing, *7th Superconducting SFQ VLSI Workshop* (2014).
- [13] Tanaka, M., Yamanashi, Y., Irie, N., Park, H., Iwasaki, S., Takagi, K., Taketomi, K., Fujimaki, A., Yoshikawa, N., Terai, H. et al.: Design and implementation of a pipelined 8 bit-serial single-flux-quantum microprocessor with cache memories, *Superconductor Science and Technology*, Vol.20, No.11, p.S305 (2007).
- [14] Volkmann, M.H., Sahu, A., Fourie, C.J. and Mukhanov, O.A.: Implementation of energy efficient single flux quantum digital circuits with sub-aJ/bit operation, *Superconductor Science and Technology*, Vol.26, No.1, p.015002 (online), available from <http://stacks.iop.org/0953-2048/26/i=1/a=015002> (2013).
- [15] Yamanashi, Y., Nishigai, T. and Yoshikawa, N.: Study of LR-Loading Technique for Low-Power Single Flux Quantum Circuits, *IEEE Trans. Applied Superconductivity*, Vol.17, No.2, pp.150–153 (online), DOI: 10.1109/TASC.2007.898608 (2007).
- [16] Yamanashi, Y., Tanaka, M., Akimoto, A., Park, H., Kamiya, Y., Irie, N., Yoshikawa, N., Fujimaki, A., Terai, H. and Hashimoto, Y.: Design and Implementation of a Pipelined Bit-Serial SFQ Microprocessor, *IEEE Trans. CORE 1β, Applied Superconductivity*, Vol.17, No.2, pp.474–477 (2007).
- [17] 田中雅光：単一磁束量子回路に基づく超高速マイクロプロセッサに関する研究，博士論文，名古屋大学 (2006)。
- [18] 寺井弘高，山下太，三木茂人，牧瀬圭正，王 鎮：超伝導単一磁束量子 (SFQ) 論理回路，低温工学，Vol.48, No.2, pp.77–82 (オンライン)，DOI: 10.2221/jcsj.48.77 (2013)。
- [19] 朴 熙中，山梨裕希，吉川信行，田中雅光，藤巻 朗，寺井弘高，萬 伸一：SFQ論理回路を用いた4-bビット・

スライス・アダーの設計, 電子情報通信学会技術研究報告, SCE, 超伝導エレクトロニクス, Vol.106, No.503, pp.13–18 (2007).



石田 浩貴 (学生会員)

1993 年生. 2016 年九州大学工学部電気情報工学科卒業. 2016 年同大学大学院修士課程入学, 現在に至る. 単一磁束量子マイクロプロセッサ・アーキテクチャに関する研究に従事. 電子情報通信学会, IEEE 各会員.



田中 雅光

1979 年生. 2006 年名古屋大学大学院電子情報学専攻修了. 博士 (工学). 2009 年より同大学院情報科学研究科特任助教. 2016 年より同大学院工学研究科助教, 現在に至る. 単一磁束量子集積回路の応用および設計技術に関する研究に従事. 電子情報通信学会, 低温工学・超伝導学会, 電気学会, 応用物理学会, IEEE 各会員.



小野 貴継 (正会員)

1982 年生. 2009 年九州大学大学院システム情報科学府情報理学専攻博士課程修了. 博士 (工学). 同年より日本学術振興会特別研究員 (PD). 2010 年株式会社富士通研究所入社. データセンタ向けサーバの研究開発に従事. 2015 年より九州大学大学院システム情報科学研究院助教, 現在に至る. メモリ・アーキテクチャ, スーパーコンピューティング等に関する研究に従事. 電子情報通信学会, IEEE 各会員.



井上 弘士 (正会員)

1971 年生. 1996 年九州工業大学大学院情報工学研究科修士課程修了. 同年横河電機 (株) 入社. 1997 年より (財) 九州システム情報技術研究所研究助手. 1999 年の 1 年間 Halo LSI Design & Device Technology, Inc. において訪問研究員としてフラッシュ・メモリの開発に従事. 博士 (工学). 2001 年より福岡大学工学部電子情報工学科助手. 2004 年より九州大学大学院システム情報科学研究院助教授. 2007 年より同大学准教授. 2014 年より同大学教授, 現在に至る. 高性能/低消費電力プロセッサ/メモリ・アーキテクチャ, セキュア・アーキテクチャ, スーパーコンピューティング, ポストムーア・コンピューティング, 等に関する研究に従事. 電子情報通信学会, ACM, IEEE 各会員.