

アーキテクチャ探索を目的とした 単一磁束量子回路の電力効率モデリング

福光 孝介^{1,a)} 石田 浩貴¹ 長岡 一起² 田中 雅光² 川上 哲志¹ 谷本 輝夫¹ 小野 貴継¹
藤巻 朗² 井上 弘士¹

概要: 近年, CMOS デバイスを用いたマイクロプロセッサは消費電力の問題により, 動作周波数の向上が頭打ちとなっている. また, サーバ等の大規模システムでは消費電力の増大が深刻な問題となっており, 消費電力あたりの性能の向上も重要視されている. そこで我々は, これらの問題を解決するために高速性, 低消費電力性を持つデバイスである単一磁束量子 (SFQ) 回路に着目した研究を進めている. 本稿では, SFQ 回路を対象としたアーキテクチャ探索を可能とするための電力効率モデルを構築する. 実設計データとの比較に基づく精度検証を行った結果, 誤差率は 10~20%程度であることが分かった.

1. はじめに

コンピュータの主要構成要素であるマイクロプロセッサは, トランジスタの微細化を拠り所として継続的な性能向上を実現してきた. しかしながら, トランジスタの微細化に伴いリーク消費電力が指数関数的に増加し, デナード則 [1] が成り立たない状況となった. また, 回路の大規模化や高速化に伴い消費電力は増加の一途をたどり, チップ発熱問題がより深刻化した. その結果, 近年のマイクロプロセッサにおいては, 消費電力の増加を回避すべく動作周波数を抑制せざるを得ない状況にある. さらに, サーバやスーパーコンピュータでは消費電力の増大が深刻な問題となっている. そのため, より高速で低消費電力な計算機の開発は必要不可欠である.

そこで我々は, 単一磁束量子 (Single Flux Quantum: SFQ) 回路 [2] に着目した研究開発を進めている. 実際に, SFQ 回路を用いておよそ 10TOPS/W で動作する乗算器の設計試作が報告されている [3]. また, さらなる電力効率の改善を目的に低電圧駆動 SFQ 回路の研究が進められており, 電源電圧 (バイアス電圧) を低くすることで動作周波数が低下するものの電力効率を大きく改善できることが報告されている [4]. しかしながら, 現状はチップ試作例が示された段階であり, その電力効率特性解析はジョセフソン素子レベルに留まっている. 今後, 低バイアス電圧化技術を演算器やプロセッサといった大規模 SFQ 回路へ適用する

には, より抽象度の高いレベルでの電力効率解析が必要となる.

そこで本研究では, SFQ 回路に用いられる主なクロッキング方式や論理ゲートが満たすべきタイミング制約を考慮し, SFQ 回路での消費電力ならびに動作周波数モデルを導出する. そして, 積和演算器を対象とした実設計結果に基づくモデルの妥当性検証を行う.

本稿の構成は次の通りである. 第 2 節では SFQ 回路の特徴, ならびに動作原理について述べる. 第 3 節では本研究で検討した SFQ 回路の周波数, 電力効率モデルについて述べる. 第 4 節では検討した電力効率モデルの検証を行う. 最後に第 5 節でまとめる.

2. 単一磁束量子回路

本節では, SFQ 回路の基本デバイスであるジョセフソン接合とその動作原理, SFQ 回路の論理構成法とクロッキング方式について説明する.

2.1 ジョセフソン接合

ジョセフソン接合 (Josephson Junction: JJ) は SFQ 回路の基本デバイスであり, 二つの超伝導体間に絶縁体または常伝導体の薄い障壁層を挟んだ図 1(a) のような構造を採る [5]. 本稿で対象とするジョセフソン接合は, 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とし, ヒステリシス特性を消すための適切な値の抵抗が超伝導電極の両端に接続されているとする.

図 1(b) にジョセフソン接合の電気特性を示す. 縦軸が

¹ 九州大学

² 名古屋大学

^{a)} kosuke.fukumitsu@cpc.ait.kyushu-u.ac.jp

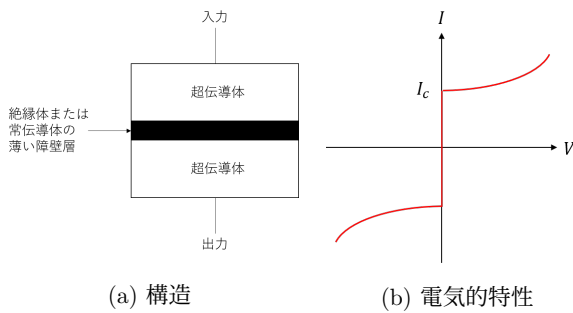


図 1: ジョセフソン接合の構造と電気的特性

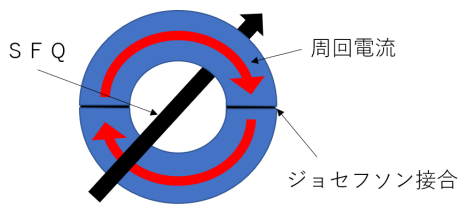


図 2: ジョセフソン接合を含む超伝導のループ

ジョセフソン接合に流れる電流，横軸がジョセフソン接合の両端（障壁層で隔たれた二つの超伝導体）の電位差を示している。電位差がない状態での最大電流値を臨界電流値 I_c と呼び、 I_c を超えると接合の両端に電位差が生じる。この電圧が発生する状態になることを接合が切れる，またはスイッチすると言う。

2.2 動作原理

SFQ 回路は図 2 の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。超伝導体ループの中に存在する磁束は連続的な値を取ることができず、磁束 Φ は $\Phi_0 = 2.07 \times 10^{-15} \text{ Wb}$ の整数倍に量子化される。

SFQ 回路では、磁束量子 Φ_0 を情報担体として用いる。具体的には、超伝導体ループ内に磁束量子が存在する場合に '1'、存在しない場合に '0' のビット情報を保持しているとみなす。たとえば、図 2 のようにループ内に磁束量子が存在する状態を考えると、ループには手前からみて時計回りに周回電流が流れる。ループ一周のインダクタンスを L とすると、2つのジョセフソン接合に周回電流 $I_L (\approx \Phi_0/L)$ が流れる。 I_L が臨界電流値 I_c を下回る場合には磁束量子が超伝導体ループに留まる。つまり、ビット情報 '1' を保持し続ける。一方、 I_L が I_c を上回る場合は接合が切れ、ジョセフソン接合間に電位差が生じる。その際、ループ内の磁束量子はループの外に出る。このとき、ループ内の磁束量子は失われ、ビット情報 '0' となる。このようにジョセフソン接合のスイッチ制御により、磁束量子の保持や伝搬が可能になる。

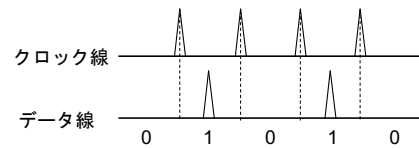


図 3: クロック同期式の概念図

2.3 パルス論理

ジョセフソン接合がスイッチすると、磁束量子の移動に伴い接合の両端に SFQ パルスと呼ばれるインパルス状の電圧が発生する。CMOS 回路は電圧レベルの 'High', 'Low' を '1', '0' と対応させる「レベル論理」であるのに対し、SFQ 回路では、磁束通過時に発生する微弱な SFQ パルスの有無で '1', '0' を表す「パルス論理」を用いて論理演算を行う。パルス論理ではレベル論理とは異なり、単体パルスでは論理状態を表現できない。たとえば、'1' をパルスの到着で表すとすると、'0' とパルスが到着していない状態の区別ができない。つまり、'0' を扱うための工夫が必要となる。代表的な手法はクロック同期式と呼ばれる論理構成法であり、その概要を図 3 に示す。この方法では、データ線とクロック線の二つを用意し、クロックパルスが論理ゲートに到着するまでにデータパルスが到着すれば '1'、到着しなければ '0' と判別する。

2.4 クロッキング方式

SFQ 回路では、回路を構成する論理ゲートにクロック信号を入力する必要がある。分配方式や回路のフィードバックループの有無によって回路の性能や動作は大きく異なる。本節では、主に用いられているコンカレント・フロー方式、カウンター・フロー方式、バイナリ・ツリー方式、ならびに、コンカレント・フロー方式とカウンター・フロー方式を組み合わせたブランチ方式について説明する。

2.4.1 コンカレント・フロー方式

対象回路においてフィードバックループが無い/有る場合におけるコンカレント・フロー方式の概要を図 4(a)/(b) に示す。SFQ 回路において、単一の SFQ パルス信号を異なる複数の通信路へ伝搬させるためにはスプリッタ (図 4 の黒点) が必要となる。コンカレント・フロー方式は、データの伝搬方向とクロック信号の伝搬方向が同一である。したがって、次の論理ゲートにデータラインよりも先にクロック信号が到着するように設計する。コンカレント・フロー方式は、クロック信号をデータが追いかけるように動作するため、2つの信号の到着時刻の差が短くなる。そのため、配線遅延を打ち消すことができ、周波数を高くすることが可能である。しかしながら、タイミング制約を満たすための条件も厳しくなるため、精密なタイミング設計が必要である。また、回路がフィードバックループを持つと 2つの信号の到着時刻の差が長くなるため、配線遅延が大きくなり、周波数が下がってしまうという欠点もある。

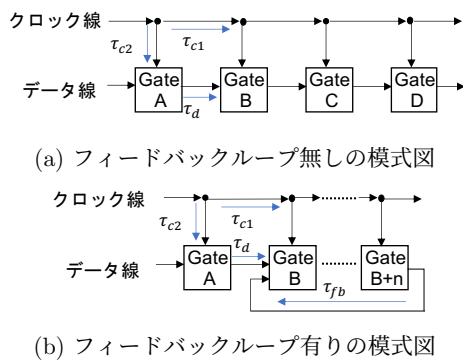


図 4: コンカレント・フロー方式の模式図

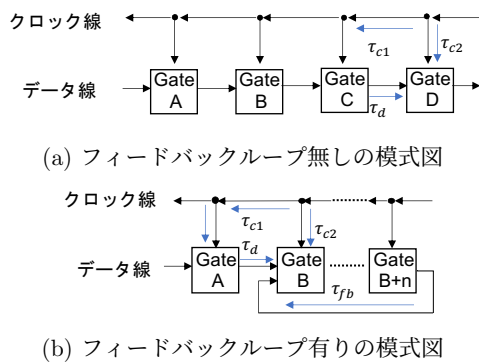


図 5: カウンター・フロー方式の模式図

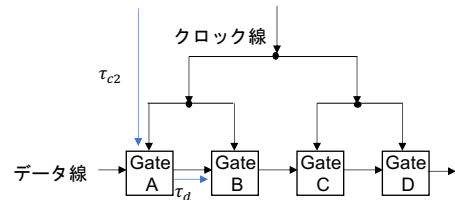
2.4.2 カウンター・フロー方式

対象回路においてフィードバックループが無い/有る場合におけるカウンター・フロー方式の概要を図 5(a)/(b) に示す。カウンター・フロー方式は、データとクロック信号の伝搬方向が逆方向であり、データの受け手の論理ゲートからデータが出力された後に、送り手がデータを渡す動作になる。

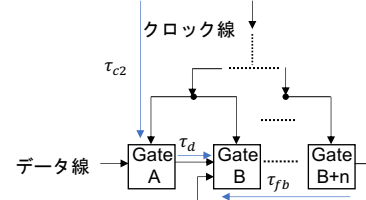
カウンター・フロー方式は、十分にクロック信号の間隔を開ければ配線遅延によりタイミングの制約は満たすため、タイミング設計が容易である。しかしながら、クロック信号とデータは伝搬方向が逆方向であるため、配線遅延は大きくなるため、コンカレント・フロー方式やバイナリ・ツリー方式と比べ、周波数を上げることができない。一方、回路がフィードバックループを持つとき、クロック信号とフィードバックされるデータの伝搬方向は順方向であるため、2つの信号の到着時刻の差は短くできる。そのため、配線遅延は大きくならず、周波数への影響は大きくないという利点もある。

2.4.3 バイナリ・ツリー方式

対象回路においてフィードバックループが無い/有る場合におけるバイナリ・ツリー方式の概要を図 6(a)/(b) に示す。バイナリ・ツリー方式は、すべての論理ゲートが同じタイミングでクロック信号を受け取る。そのため、クロック信号が同時に到着するので設計において理解しやすいが、クロック配線のコストが非常に大きいため現実的では



(a) フィードバックループ無し



(b) フィードバックループ有り

図 6: バイナリ・ツリー方式の模式図

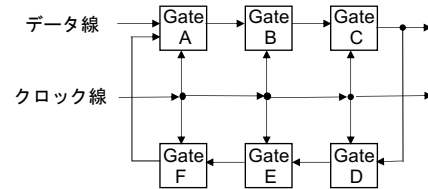


図 7: ブランチ方式の模式図

ない。

2.4.4 ブランチ方式

ブランチ方式は、コンカレント・フロー方式とカウンター・フロー方式を組み合わせたハイブリッドなクロッキング方式である。ブランチ方式の模式を図 7 に示す。図 7 の論理ゲート A~C はデータとクロック信号の伝搬方向が順方向であるコンカレント・フロー方式であり、論理ゲート D~F はデータとクロック信号の伝搬方向が逆方向カウンター・フロー方式である。また、データがコンカレント・フロー方式の部分からカウンター・フロー方式の部分に移動する箇所（論理ゲート C から D）とデータがカウンター・フロー方式の部分からコンカレント・フロー方式の部分に移動する箇所（論理ゲート F から A）は、クロック信号が同時に入力されるため、バイナリ・ツリー方式であるとみなせる。

ブランチ方式の動作周波数は、コンカレント・フロー方式の部分、カウンター・フロー方式の部分、バイナリ・ツリー方式の部分の最大遅延により決まる。そのため、周波数を上げるためには、比較的複雑な処理を行う部分は比較的高い周波数となるコンカレント・フロー方式で行う必要がある。

3. SFQ 回路の電力効率モデリング

本節では、本研究で提案する電力効率、電力、動作周波数、ジョセフソン接合数のモデルについて説明する。

3.1 消費電力モデル

SFQ 回路の全体消費電力 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 (1) で表される。

$$P_{all} = P_{dynamic} + P_{static} \quad (1)$$

$$P_{dynamic} \simeq \alpha \Phi_0 I_c f \times J_{all} \quad (2)$$

$$P_{static} = V_{bias} \times I_{bias} \times J_{all} \quad (3)$$

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_{bias} はバイアス電圧、 I_{bias} は一つのジョセフソン接合に供給するバイアス電流、 J_{all} はジョセフソン接合数である。なお、1 個のジョセフソン接合のスイッチ動作あたりの動的消費電力は $\Phi_0 I_c$ のオーダーである。

3.2 動作周波数モデル

まず、通常電圧時の動作周波数を考える。SFQ 回路の周波数はクロッキング方式と回路構成に大きく依存する。本研究での SFQ 回路はゲートレベル・パイプライン構造 [6] を想定している。これは、論理ゲート 1 段で 1 つのステージを構成する細粒度パイプライン構造である。

クロック同期式の SFQ 回路において、データ入力後にクロック信号の入力が禁止される時間を SetupTime、クロック信号入力後のデータの入力が禁止される時間を HoldTime という。SetupTime と HoldTime はバイアス電圧に依存しており、バイアス電圧が低くなるほど、これらは長くなる。また、これらは論理ゲートの種類によっても異なる。たとえば、D フリップフロップや AND ゲートなどでは Setup/HoldTime は短く、XOR ゲートなどでは比較的長くなる。

SFQ 回路には上記の通り、SetupTime と HoldTime により満たさなければならないタイミング制約が 2 つ存在する。1 つは、クロック信号の入力時刻から HoldTime 以上の時間が経過した後にデータを入力する必要がある。もう 1 つは、データが入力された時刻から SetupTime 以上の時間が経過した後にクロック信号を入力する必要がある。これら 2 つの制約は式 (4)、(5) ならびに図 8 で表現される。

$$T_{clock} + HoldTime \leq T_{data} \quad (4)$$

$$T_{data} + SetupTime \leq T_{clock} + CCT \quad (5)$$

ここで T_{clock} はクロック信号の入力時刻、 T_{data} はデータの入力時刻、 CCT (Clock Cycle Time) はクロック信号の周期である。このとき、SFQ 回路の動作周波数は式 (6) で表される。

$$f = \frac{1}{SetupTime + HoldTime + \Delta\tau} \quad (6)$$

$\Delta\tau$ は回路による追加遅延であり、回路構成、クロッキング

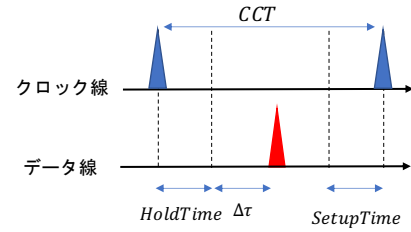


図 8: SetupTime と HoldTime による制約の模式図。青の三角がクロック信号、赤の三角がデータを表す。

方式に依存する値である。また、回路が制約を満たすために $\Delta\tau$ が 0 以上の値となるように設計をする必要がある。

次に、クロッキング方式による $\Delta\tau$ の値について説明をする。たとえば、図 4(a) のフィードバックループのないコンカレント・フロー方式でのデータとクロック信号のタイミングを論理ゲート B に着目する。論理ゲート A に入力するためのクロック線の分岐点における時刻を $T = 0$ とおくと、クロック信号が論理ゲート A に入力された後に、論理ゲート A から出力されたデータが論理ゲート B に入力される時刻 T_{data} は $\tau_d + \tau_{c2}$ となる。また、論理ゲート B にクロック信号が入力される時刻 T_{clock} は $\tau_{c1} + \tau_{c2}$ となる。これらの値を式 (4)、(5) に代入し、整理すると満たすべき条件は以下の式 (7) で表される。

$$\begin{aligned} SetupTime + HoldTime \\ &\leq SetupTime + (\tau_d - \tau_{c1}) \quad (7) \\ &\leq CCT \end{aligned}$$

この条件式 (7) と式 (6) よりコンカレント・フロー方式の $\Delta\tau$ の値は、以下の式 (8) で表される。

$$\Delta\tau = (\tau_d - \tau_{c1}) - HoldTime \quad (8)$$

同様にそれぞれのクロッキング方式で論理ゲートに対してデータとクロック信号のタイミングを考えると $\Delta\tau$ の値は以下ようになる。

$$\Delta\tau = \begin{cases} \tau_d - \tau_{c1} - HoldTime & (\text{concurrent}) \\ \tau_d + \tau_{c1} - HoldTime & (\text{counter}) \\ \tau_d - HoldTime & (\text{binarytree}) \end{cases} \quad (9)$$

また、フィードバックループがある場合を考えると、フィードバック部分の $\Delta\tau$ の値は以下ようになる。

$$\Delta\tau_{fb} = \begin{cases} \tau_{fb} + n\tau_{c1} - HoldTime & (\text{concurrent}) \\ \tau_{fb} - n\tau_{c1} - HoldTime & (\text{counter}) \\ \tau_{fb} - HoldTime & (\text{binarytree}) \end{cases} \quad (10)$$

ここで、 τ_{fb} はフィードバックでの伝搬時間、 n はフィードバック段数である。また、ブランチ方式は、コンカレント・フロー方式とカウンター・フロー方式とバイナリ・ツリー方式を含んだ回路となっているため、ブランチ方式の $\Delta\tau$ の値は、以下の式 (11) で表される。

$$\Delta\tau_{branch} = \max\{\tau_{concurrent}, \tau_{counter}, \tau_{binarytree}\} \quad (11)$$

次に低バイアス電圧時を考える。SFQ回路による論理演算は複数のSFQパルスによる相互作用で実現されるため、SFQパルス同士の待ち合わせ時間が必要となる。このため、SFQパルス幅の数倍程度がクロック信号の最小間隔となる。SFQパルスの幅は、ジョセフソン接合がスイッチする際に接合の両端に発生する電圧値 (V_{peak} と呼ぶ) に反比例して小さくなる。これはSFQパルスの時間積分は磁束量子 Φ_0 で一定となるためである。 V_{peak} の最大値は接合の臨界電流値 I_c と抵抗 R の積に相当し、製造プロセスによって決まる一定値となる。また、磁束量子はパルス幅と V_{peak} の積で近似され、 V_{peak} は V_{bias} にも依存をしている [7]。 V_{bias} が $I_c R$ より高い場合、 V_{peak} は $I_c R$ で表される。一方、 V_{bias} が $I_c R$ より低い場合、 V_{peak} は V_{bias} に比例する。そのため、低バイアス電圧時を考慮したSFQ回路の動作周波数は式 (6) と表す。

$$f = \frac{1}{k(V_{bias}) \times \Delta t}$$

$$k(V_{bias}) = \begin{cases} \frac{SetupTime+HoldTime+\Delta\tau}{\Delta t_0} \\ \frac{SetupTime+HoldTime+\Delta\tau}{\Delta t_0} - 1(lowVoltage) \end{cases} \quad (12)$$

$$\Delta t = \max\left\{\frac{\Phi_0}{V_{bias}}, \Delta t_0\right\}$$

ここで、 Δt はSFQパルス幅、 Δt_0 はSFQパルス幅の下限 (プロセス依存)、 $k(V_{bias})$ は回路構成、 V_{bias} に依存する値である。

3.3 ジョセフソン接合数モデル

3.3.1 基本モデル

SFQ回路のジョセフソン接合数は式 (13) で表される。

$$JJ_{all} = JJ_{logic} + JJ_{wire} \quad (13)$$

JJ_{logic} はSFQ論理ゲートを構成するジョセフソン接合数である。 JJ_{wire} は論理ゲート間の配線やクロックラインによるジョセフソン接合数で式 (14) と表す。

$$\begin{aligned} JJ_{wire} = & inputGate \times JJ_{driver,receiver} \\ & + (n_{Gate}/2 + n_{pipeline}) \times JJ_{driver,receiver} \\ & + n_{clock-splitter} \times JJ_{splitter} \end{aligned} \quad (14)$$

ただし、 $inputGate$ はSFQ回路に含まれる全論理ゲートの入力数の和、 $JJ_{driver,receiver}$ は配線のドライバとレシーバによるJJ数、 n_{Gate} は総SFQ論理ゲート数、 $n_{pipeline}$ はパイプライン段数、 $n_{clock-splitter}$ はクロックラインで用いられるスプリッタ数、 $JJ_{splitter}$ はスプリッタに用いられるJJ数である。配線部分はドライバとレシーバ

のみJJで構成されているため、論理ゲート間の配線は、 $inputGate \times JJ_{driver,receiver}$ で表されている。

また、 $n_{clock-splitter}$ は式 (15) で表す。

$$n_{clock-splitter} = \log_3 \left[\frac{n_{Gate}}{n_{pipeline}} \right] \quad (15)$$

3.3.2 Brent-Kung Adder ベースの並列乗算器の場合

本節では、乗算器のジョセフソン接合数を見積もるために乗算器に用いられるSFQ論理ゲート数を算出する。

本稿で想定する乗算器では2つの入力各ビットの論理積をとる部分積生成部、ビットの各桁への入力数が2つ以下になるまで加算を行う桁上げ加算部、桁上げ条件を生成する桁上げ条件生成部、並列して桁上げを行う並列加算部、各ビットの積和を求める和計算部から構成される。これらの論理ゲート数をそれぞれ見積もる。

まず、部分積生成部の論理ゲート数を考える。部分積生成部のパイプライン段数は1段で、ANDゲートによって構成され式 (16) で表される。

$$AND(n_{bit}) = n_{bit}^2 \quad (16)$$

次に、桁上げ加算部の論理ゲート数を算出する。桁上げ加算部の1段目の i ビット目への入力数 n_i は式 (17) で表される。

$$n_i = \frac{n_{bit}}{2} - \left| i - \frac{n_{bit}}{2} + 1 \right| \quad (17)$$

ここで、 i はビットの位である。SFQ回路での桁上げ加算では全加算器と半加算器、D-FFが用いられる。このとき、各パイプラインステージの各ビットでのこれらの数は (18) で表される。

$$\begin{aligned} FA(n) &= \left\lceil \frac{n}{3} \right\rceil \\ HA(n) &= \left\lceil \frac{n - 3f(n)}{2} \right\rceil \end{aligned} \quad (18)$$

$$DFF(n) = 2(n - 3f(n) - 2h(n))$$

ここで n は各ビットへの入力数、 $f(n)$ 、 $h(n)$ 、 $d(n)$ はそれぞれ各ビットでの全加算器、半加算器、D-FFの数である。また、次のパイプラインステージの i ビット目への入力数 n_{i-next} は式 (19) で表される。

$$\begin{aligned} n_{i-next} = & FA(n_i) + HA(n_i) + DFF(n_i) \\ & + FA(n_{i-1}) + HA(n_{i-1}) \end{aligned} \quad (19)$$

桁上げ加算部では、 n_{i-next} が2以下になるまで桁上げ加算を行う。

次に、条件生成部の論理ゲート数を考える。条件生成部のパイプライン段数は1段で、XORゲートとANDゲート、D-FFによって構成され、これらの数は (20) で表される。

$$\begin{aligned}
 XOR(n_{bit}) &= k - 1 \\
 AND(n_{bit}) &= k \\
 DFF(n_{bit}) &= 2n_{bit} - k \\
 k &: \sum_{i=1}^{n_{bit}} (n_{i-next} - 1)
 \end{aligned} \tag{20}$$

次に並列加算部について考える。並列加算部では、桁上げ信号を順次計算せず並列化してまとめて加算をする方法である。今回の見積もりでは、配線密度やファンアウトなどのSFQ回路の設計制約を考慮し、並列プリフィクス加算器のBrent-Kung Adder [8]の構造を参考にした。

並列加算部はD-FFとPB(Prefix Box)と呼ばれるものから構成される。PBが各ビットの桁上げの特微量を伝搬することにより並列加算を行うことができる。並列加算部は入力されるビット数によりパイプライン段数、PB数が求まる。また、PBは2個のANDゲートと1個のWired OR、3個のスプリッタから構成される。並列加算部のパイプライン段数は式(21)で表される。

$$\begin{aligned}
 &[\log_2(n-1)] + \max\{[\log_2(n-1)] - 1, [\log_2 m] + 1\} \\
 n &: \text{入力数が2以上のビット数} \\
 m &: n - 2^{\lceil \log_2(n-1) \rceil}
 \end{aligned} \tag{21}$$

また、Prefix Boxの個数は式(22)で表される。

$$\begin{aligned}
 &\sum_{i=1}^{\lceil \log_2 n \rceil} (2^i - 1) + \sum_{j=1}^l \left\lceil \frac{m}{2^{j-1}} \right\rceil \\
 n &: \text{入力数が2以上のビット数} \\
 m &: n - 2^{\lceil \log_2 n \rceil} \\
 l &: \lceil \log_2 \max\{m, 1\} \rceil + 1
 \end{aligned} \tag{22}$$

最後に和計算部について考える。和計算部のパイプライン段数は1段で、XORゲートとD-FFによって構成され、これらの数は(23)で表される。

$$\begin{aligned}
 XOR(n) &= k \\
 DFF(n) &= 2n_{bit} - k
 \end{aligned} \tag{23}$$

式(16)~(23)により、総ゲート数 n_{Gate} 、パイプライン段数 $n_{pipeline}$ 、SFQ論理ゲートへの総入力数 $input_{Gate}$ が求まる。

3.4 電力効率モデル

本研究で導入電力効率モデルが前提とする単位は、消費電力あたりの演算実行数(OPS/W)とする。OPS(Operations Per Second)は1秒あたりの平均演算実行数であり、動作周波数と1クロックサイクルあたりの演算実行数(Operations Per Cycle: OPC)の積と定義される。OPC

は同時演算実行数 ω とパイプラインストール率 γ 、パイプライン段数 p を用いて式(24)で表される。

$$OPC = \omega / (1 + \gamma \times (p - 1)) \tag{24}$$

γ はパイプラインステージのストール率であり、 $\gamma = 1$ の場合は $(p-1)$ 段のパイプラインステージ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。今回のSFQ回路ではパイプラインストールは発生しない回路を想定しているため、パイプラインストール率 $\gamma = 0$ とした。また、バイアス電流 I_B と臨界電流値 I_c の最適比は $I_B/I_c = 0.70$ であることが示されている[9]。そのため、電力効率モデルは式(25)で表される。

$$\begin{aligned}
 OPS/W &= \frac{OPC \times f}{P_{all}} \\
 &= \frac{\omega \times f}{(\alpha \Phi_0 f + 0.7V_{bias}) \times I_c N_{JJ}}
 \end{aligned} \tag{25}$$

4. 電力効率モデルの検証

本節では、実設計データに基づくSFQ電力効率モデルの精度評価を行い、その妥当性を検証する。本検証で用いる3種類のSFQ回路を表1に示す。ここで、表中の「クリティカル論理ゲート」とは、各SFQ回路において、SetupTimeとHoldTimeの和が最大となるクリティカルパス上の論理ゲートを指す。なお、4×4+8ビットの積和演算器はフィードバックループを有するため、クロッキングにブランチ方式を採用している。そのため、コンカレント・フロー回路部分とカウンター・フロー回路部分が存在するが、クリティカルパスは後者に存在する。

各SFQ回路において表2の値と3.3節の式を用いてジョセフソン接合数を見積ると、8bitMultiplierは14,786個、4bitMultiplierは4,027個、4bitMACは7,435個であり、提案モデルを用いた見積り値、ならびに、実設計との誤差は表5となった。4bitMultiplierと4bitMACでは誤差率が10%程度であり、比較的高い精度で電力性能を見積もることができている。一方、8bitMultiplierでは誤差率が20%程度と大きい。この原因としては、動作周波数モデルでは論理ゲート間の配線が理想に近いものと考えているため、回路規模が大きくなり配線が複雑になることによる動作周波数の変化を正確に反映できていないことが考えられる。

5. おわりに

本稿では、SFQ回路の電力効率を推定するためのモデルを考案した。実設計データとの比較を行った結果、電力効率推定誤差は10~20%程度であった。なお、本稿で提案したモデルはSFQ回路技術に基づくニューラルネットワーク・アクセラレータのアーキテクチャ探索に用いられている[10]。

表 1: 検証用 SFQ 回路

回路	バイアス電圧	クロッキング方式	クリティカル論理ゲート	モデル
8bitMultiplier [3]	通常電圧	コンカレント・フロー	XOR	式 (9), (12)
4bitMultiplier	低電圧	コンカレント・フロー	XOR	式 (9), (12)
4bitMAC	低電圧	ブランチ	D-FF	式 (9), (12)

表 2: SFQ 論理ゲートの情報

論理ゲート	JJ count	Delay ps	SetupTime ps	HoldTime ps
D-FF	6	5.1	1.2	-0.9
AND	14	7.9	-1.8	2.7
XOR	11	6.5	3.7	4.1
Splitter	3	4.3	n/a	n/a
WiredOR	7	8.2	n/a	n/a

表 3: 実チップの値

	V_{bias} mV	frequency GHz	JJ count
4bitMultiplier	0.46	52	4,498
4bitMAC	0.53	38	9,739
8bitMultiplier	2.5	48	20,251

表 4: 電力効率算出に用いるパラメータ

パラメータ	値
α	0.5(-)
Δt	2.0(ps)

表 5: モデルと実データの比較

	4bitMultiplier		4bitMAC		8bitMultiplier	
	実測	モデル	実測	モデル	実測	モデル
f[GHz]	52	47.3	38	42.0	48	66.6
誤差率	+9.98%		-9.47%		-28.0%	
power[μ W]	134	151	366	365	5,600	4935
誤差率	-11.0%		+0.24%		+13.4%	
TOPS/W	381	314	104	115	10	13.5
誤差率	+21.3%		-9.53%		-26.0%	

今後の課題としては、低電圧時の SetupTime と Hold-Time の値を考慮したモデルの詳細化、ならびに、超伝導状態で動作するための冷却電力の考慮が挙げられる。

謝辞 本研究を進めるにあたり、活発な議論とご協力を頂いた九州大学井上研究室のみなさまに心より感謝の意を表します。なお、本研究は、JST、未来社会創造事業、JPMJMI18E1、ならびに、一部文部科学省科学研究費補助金 JP19H01105, JP18H05211, JP18J21274 の支援を受けたものである。

参考文献

[1] Dennard, R. H., Gaensslen, F. H., Yu, H., Rideout, V. L.,

Bassous, E. and LeBlanc, A. R.: Design of ion-implanted MOSFET's with very small physical dimensions, *IEEE Journal of Solid-State Circuits*, Vol. 9, No. 5, pp. 256–268 (online), DOI: 10.1109/JSSC.1974.1050511 (1974).

[2] Likharev, K. K. and Semenov, V. K.: RSFQ logic/memory family: A new Josephson-junction technology for sub-terahertz-clock-frequency digital systems, *IEEE Transactions on Applied Superconductivity*, Vol. 1, No. 1, pp. 3–28 (1991).

[3] Nagaoka, I., Tanaka, M., Inoue, K. and Fujimaki, A.: 29.3 A 48GHz 5.6 mW Gate-Level-Pipelined Multiplier Using Single-Flux Quantum Logic, *2019 IEEE International Solid-State Circuits Conference-(ISSCC)*, IEEE, pp. 460–462 (2019).

[4] 田中雅光, 長岡一起, 石田浩貴, 佐野京祐, 山下太郎, 小野貴継, 井上弘士, 藤巻朗: 単一磁束量子回路に基づくゲートレベル・パイプライン算術論理演算器の高エネルギー効率化と 0.3 mW, 30GHz 動作実証, 研究報告システム・アーキテクチャ (ARC), Vol. 2019, No. 3, pp. 1–8 (2019).

[5] 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).

[6] 石田浩貴, 田中雅光, 小野貴継, 井上弘士: 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, 情報処理学会論文誌, Vol. 58, No. 3, pp. 629–643 (2017).

[7] Tanaka, M., Ito, M., Kitayama, A., Kouketsu, T. and Fujimaki, A.: 18-GHz, 4.0-aJ/bit Operation of Ultra-Low-Energy Rapid Single-Flux-Quantum Shift Registers, *Japanese Journal of Applied Physics*, Vol. 51, p. 053102 (online), DOI: 10.1143/jjap.51.053102 (2012).

[8] Brent and Kung: A Regular Layout for Parallel Adders, *IEEE Transactions on Computers*, Vol. C-31, No. 3, pp. 260–264 (online), DOI: 10.1109/TC.1982.1675982 (1982).

[9] 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 (2006).

[10] Ishida, K., Byun, I., Nagaoka, I., Fukumitsu, K., Tanaka, M., Kawakami, S., Tanimoto, T., Ono, T., Kim, J. and Inoue, K.: SuperNPU: An Extremely Fast Neural Processing Unit Using Superconducting Logic Devices (2020).