

非同期単精度浮動小数点除算器の方式検討とFPGA実装

廣本 正之[†] 高橋 温子[†] 神山 真一[†] 越智 裕之[†] 中村 行宏^{††}

[†] 京都大学大学院 情報学研究科 通信情報システム専攻, 〒606-8501 京都市左京区吉田本町
^{††} 立命館大学 総合理工学研究機構, 〒525-8577 滋賀県草津市野路東 1-1-1
E-mail: [†]reconf@easter.kuee.kyoto-u.ac.jp, ^{††}y-nakamr@fc.ritsumeimei.ac.jp

あらまし デジタルシステムの開発において単同期式設計が普及しているが、クロック周波数に応じてレジスタ間の組み合わせ回路段数を最適化しなければ十分に性能を発揮することができないため、クロック周波数が異なるシステム間での設計資産の流用は困難であった。本稿では、各々のモジュールがそれぞれ最高の速度で動作可能な非同期式システムに注目し、これをIEEE754 準拠単精度浮動小数点除算器に適用することを提案する。提案する除算器は、減算シフトの反復で仮数部の除算を行うモジュールをグローバルクロックと独立なローカルクロックで動作させ、前後の正規化や丸めのモジュールとは非同期のインターフェースで接続しようというものであり、(1) 特定のターゲットテクノロジーに対し単一の設計資産で任意のグローバルクロック周波数のシステムに対応可能、(2) ローカルクロック周波数の最適化による面積効率や電力効率の改善、などが期待される。また本稿では、提案非同期回路をXilinx社のFPGA向けに設計、評価した結果も報告する。

キーワード IP 再利用性, IEEE754, 低消費電力, 減算シフト型除算器

An Asynchronous Single-precision Floating-point Divider and its Implementation on FPGA

Masayuki HIROMOTO[†], Atsuko TAKAHASHI[†], Shin'ichi KOUYAMA[†],
Hiroyuki OCHI[†], and Yukihiro NAKAMURA^{††}

[†] Dept. of Communications and Computer Eng., Graduate School of Informatics, Kyoto Univ.,
Yoshida-Honmachi, Sakyou-ku, Kyoto 606-8501 Japan
^{††} Research Organization of Science and Engineering, Ritsumeikan Univ.,
Noji Higashi 1-1-1, Kusatsu, Shiga 525-8577 Japan
E-mail: [†]reconf@easter.kuee.kyoto-u.ac.jp, ^{††}y-nakamr@fc.ritsumeimei.ac.jp

Abstract *Synchronous* design methodology is widely used for today's digital circuits. However, highly optimized synchronous design for a specific clock frequency is difficult to be reused in another system with different clock frequency, because logic depth between FFs should be tailored for the clock frequency. In this paper, we focus on *Asynchronous* design, in which each module works at its best performance, and apply it to an IEEE754-standard single-precision floating-point divider. In our divider, a mantissa divider is driven by a high-speed local clock and connected to pre-/post-processing modules with asynchronous interface. Our divider is ready to be built into a system with arbitrary clock frequency and achieve its peak performance and area- and power-efficiency. This paper also reports an implementation result of the proposed divider on a Xilinx FPGA.

Key words IP reusability, IEEE754, low power design, digit-recurrence divider

1. 序 論

近年 VLSI の設計においては、設計の容易さや各種 CAD ツールが整っていることもあり、単同期式システムが主流となっ

ている。しかしながら同期式システムにはいくつかの問題点がある [1]。1 つは、高い周波数のグローバルクロックをチップ全域に分配する必要があるため、そのクロックツリー自体や末端に接続されているフリップフロップ群によって消費される電力

が非常に大きいことである。これは、システムによっては全体の45%に及ぶとの報告もある [2]。2つ目は、チップ内での製造ばらつきへの対応が困難なことが挙げられる。これは、プロセスの微細化に伴うトランジスタのばらつきによってもたらされた、クロックスキューの増大によるものである。このために設計マージンを大きくとる必要性が生じ、プロセスの性能を最大限に発揮することが難しくなっている。この問題に対し、製造後にばらつきを緩和する回路技術 [3] や設計方法 [4]、局所的には同期設計し大域的には非同期設計するアプローチ [5] などが提案されている。3つ目として、同期システムでは設計対象のクロック周波数毎に回路の最適化が必要であることが挙げられる。高スループットな同期式回路を実現するためにはクロック周波数に応じ、レジスタ間の組み合わせ回路の段数を最適化しなければならない。このためには、回路記述の段階でレジスタ挿入を最適に行う必要がある。しかし上流工程で配置配線後の遅延時間を精密に見積もることは困難であり、高スループットな回路の実現のためには相当の工数をかけて回路記述と遅延解析の繰り返しを行う必要がある。越智らにより開発された IEEE-754 準拠の同期式単精度浮動小数点除算器ライブラリ [6] では、動作周波数別に最適化された HDL 記述を提供しているが、このようなライブラリの開発には多大な工数を要する。

これらの問題を解決すべく、我々は非同期式システムに注目する。非同期回路の主な特徴としては以下のものがある [7]。

- (1) 低消費電力である
- (2) クロックスキューの影響を受けない
- (3) グローバルクロック周波数に依存した最適化が不要

(1) は主にクロック分配系による消費電力が削減できることによる。この特徴から、低消費電力が要求される携帯機器向け LSI への採用実績がある。(2) は、非同期式システムではグローバルクロックを用いないため、長距離配線が減りクロックスキューの影響をほとんど受けないことが期待される。(3) は、各モジュールがグローバルクロックに同期して動作する必要がないため、モジュール内の回路はグローバルクロックとは独立に設計可能なことによる。各モジュールは最も性能を発揮できる最適な回路構成を取ることができる。

本研究では、特に上の (3) に注目し、グローバルクロック周波数に依存しない高性能な演算器の設計資産の提供を試みる。ここでは、越智ら [6] が開発した IEEE-754 準拠の同期式単精度浮動小数点除算器を取り上げ、これの非同期化を行う。文献 [6] の設計では動作周波数に応じ、組み合わせ回路段数を調整したものを複数種類設計する必要があった。本稿では最も性能の良い構成で除算器の内部を設計し、外部とのインターフェースを非同期化することにより、1種類の設計のみで多様な周波数のシステムに組込める除算器を提供する。

本稿ではまず、2. 章で非同期式システムならびに浮動小数点除算について述べ、3. 章で浮動小数点除算モジュールのコアとなる、仮数部除算モジュールについて検討を行なう。その後、4. 章で提案浮動小数点除算モジュールの評価を行い、同期式と非同期式の比較を行なう。最後に、5. 章でまとめとする。

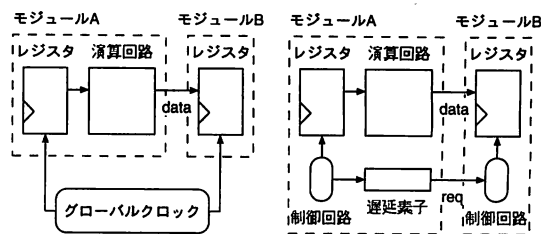


図1 同期式データ転送

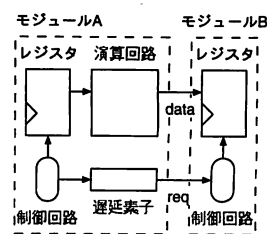


図2 非同期式データ転送

2. 単精度浮動小数点除算器の非同期化

本章では、まず非同期式システムについて述べる。次に単精度浮動小数点形式での除算について説明し、これの非同期化について述べる。

2.1 非同期式システム

本稿では、回路の逐次的な動作が全て単一のグローバルクロックに同期して行われる回路を同期式回路、そのようなグローバルクロックを持たない回路を非同期式回路と定義する。

同期式回路では図1のように、全レジスタは単一のグローバルクロックに同期してデータの送受信を行う。一方、非同期式回路ではグローバルクロックが存在しないため、データを送受信の際に各モジュール間でタイミングを決める必要がある。その方式として、データと別にタイミング信号を持たせるハンドシェイク方式や、データ1ビット毎にデータの有効・無効の情報を持たせて送る2線式などが挙げられる。ハンドシェイク方式は一度に多ビットのデータが送られるデータ系に適しており、2線式は一度に1ビットのデータが送られる制御系に適している。本稿で対象とする除算器では一度に複数ビットのデータを送るため、ハンドシェイク方式を採用する。ハンドシェイク方式では図2のように、送信側のモジュール A はデータとは別に req 信号を送信する。モジュール B は req 信号を受け取ると、データ信号線上のデータをレジスタに取り込む。データがモジュール B に到着するまで req 信号を遅らせる必要があるため、req 信号線には遅延回路が設けられる。

2.2 単精度浮動小数点除算

2.2.1 単精度浮動小数点形式

本研究で扱う浮動小数点数は、IEEE-754 準拠の単精度浮動小数点形式とする。単精度浮動小数点数で表わされる数値は、1ビットの符号フィールドを s 、8ビットの指数フィールドを e 、23ビットの仮数フィールドを f とすると、下の式の通りである [8]。

$$R = (-1)^s \times (1.f) \times 2^{e-127} \quad (1)$$

このような形式により、漸進アンダーフローを行わない場合でも、絶対値が 2^{-126} から $(1 - 2^{-24}) \times 2^{128}$ までの広範囲な数値を表現できる。

2.2.2 単精度浮動小数点除算の手順

図3に浮動小数点除算回路のブロック図を示す。浮動小数点除算の大まかな流れは次の通りである。

まず、回路に除数と被除数を入力する。それが非正規化数で

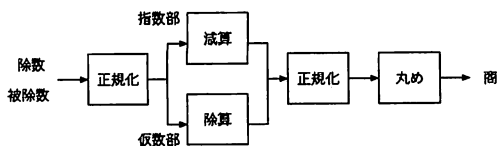


図 3 浮動小数点除算回路のブロック図

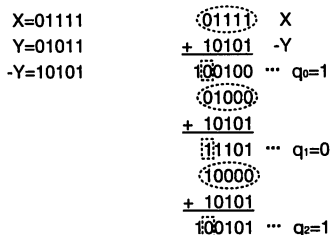


図 4 引き戻し法アルゴリズム

あった場合は、パレルシフトを用いて正規化する。次に、仮数部同士の除算と指数部同士の減算を行う。その後、正規化、丸めを行い、商を出力する。この他に、商の符号を生成したり除数が0であった場合等の例外処理を行う仕組みも必要である。

2.3 除算アルゴリズム

浮動小数点除算の中で、仮数部の除算を行うアルゴリズムについて述べる。

除算アルゴリズムは、大きく乗算型と減算シフト型に分類される。本稿では、回路規模が小さく抑えられる減算シフト型の引き戻し法を取り上げる。以下、 n 桁の減算シフトを行うものとし、被除数 X 、及び除数 Y は正規化されているものとする。また、商を $Q = \sum_{j=0}^{n-1} q_j 2^{-j}$ とし、 q_j を決定した後の部分剰余を R_{j+1} とする。

引き戻し法のアルゴリズムは、除算を筆算で行う場合の手順とほぼ同じである。 $j = 0$ の場合、 $X \geq Y$ であれば $q_0 = 1$ 、 $R_1 = X - Y$ とし、さもなければ $q_0 = 0$ 、 $R_1 = X$ とする。 $j \geq 1$ の場合、 $2R_j - Y \geq 0$ ならば $q_j = 1$ 、 $R_{j+1} = 2R_j - Y$ とし、さもなければ $q_j = 0$ 、 $R_{j+1} = 2R_j$ とする操作を繰り返す。実際の数値を用いた例を図4に示す。丸い点線で囲まれた部分が部分剰余、四角の点線で囲まれた部分が符号ビットである。 $Y = 01011$ 、 $X = 01111$ とすると、2の補数を用いて、 $-Y = 10101$ となる。まず、 $X - Y$ を計算した結果の符号ビットが0なので $q_0 = 1$ を立て、 $R_1 = X - Y$ とする。次に、 $2R_1 - Y$ 計算すると、結果の符号ビットが1であるため、 $2R_0$ を R_1 に格納する。以下同様である。引き戻し法において必要となる主なハードウェアは、 $2R_j - Y$ を計算するための減算器と、 R_j に代入する値を選択するための2入力マルチプレクサである。

2.4 浮動小数点除算器の非同期化

上述の浮動小数点除算を行う非同期回路を設計する。図3の浮動小数点除算器の中で最も回路規模や所要時間が大きいのは、仮数部同士の除算回路である。そこで本研究では、この仮数部除算回路を外部のグローバルクロックと無関係な高速なローカ

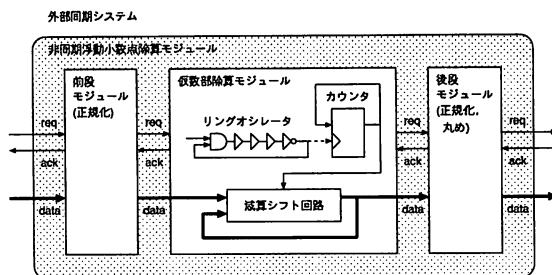


図 5 非同期浮動小数点除算モジュールの概要

ルクロックで動作させ、前後の正規化、丸めを行うモジュールとはハンドシェイク方式で接続する方式を取ることにする。非同期浮動小数点除算器の概要を図5に示す。外部とは非同期のインターフェースで接続されている。

仮数部除算モジュールは、ローカルクロック信号の生成器、24回の減算シフト回数を数えるためのカウンタ、及び減算シフト回路で構成される。仮数部除算モジュールの動作の概略は以下の通りである。図5において、前段から仮数部除算モジュールにリクエスト信号が到達するとリングオシレータが動作を開始する。リングオシレータで生成されたローカルクロック信号がカウンタや減算シフト回路に供給され、所定の回数分の減算シフトが行われる。その後、仮数部除算モジュールは次段へのリクエスト信号を立てて減算シフト結果を送り、リングオシレータの動作を停止する。

3. 仮数部除算モジュールの検討

本章では図5の非同期浮動小数点除算器において、最も回路規模の大きい部分である仮数部除算モジュールの構成について検討する。

3.1 仮数部除算モジュールの構成

引き戻し法アルゴリズムでは減算シフト操作を桁数だけ行うため、仮数部除算モジュールにおいて24回の減算シフトを行う必要がある。減算シフトを行う演算器数を N とすると、 $24/N$ 回の反復により24回の減算シフトを行うことができる。 N を大きくすると反復回数が減りスループットの向上が期待できるが、回路面積も増加するためトレードオフの関係にある。そこで N が1, 2, 3, 4, 6, 8, 12, 24の場合についてそれぞれ仮数部除算モジュールの実装を行い、回路規模、スループット、消費電力について評価を行った。

仮数部除算モジュールはリングオシレータにより生成されたローカルクロックにより動作する。リングオシレータは、減算シフトの段数 N に応じそれぞれが最も高速に動作する周波数のローカルクロックを生成するように設計した。

3.2 仮数部除算モジュールの実装及び評価

上記の仮数部除算モジュールをHDLにより記述し、シミュレーション、論理合成及びFPGAを対象とした配置配線を行った。シミュレーションにはMentor Graphics ModelSim SE 6.0cを、論理合成、配置配線にはXilinx ISE 8.1iを、ターゲットのFPGAとしてはXilinx Virtex-II XC2V1000を用いた。消

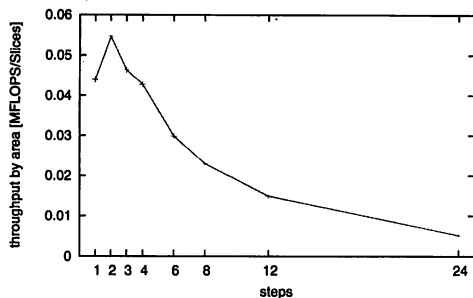


図 6 仮数部除算モジュールの回路規模あたりのスループット

費電力については、配置配線後のネットリストと遅延情報を用いたゲートレベルシミュレーションを行い、その結果を ISE に付属の消費電力解析ツール XPower を用いて見積った。

減算シフト段数を変化させた時の仮数部除算モジュールの性能を表 1 に示す。減算シフト段数が増えるに従って組み合わせ回路の遅延が大きくなるため動作周波数が低下するが、所要クロック数は少なくなる。そのためスループットは比較的似通った値となっている。回路規模については基本的には段数が増えるにつれて大きくなる傾向が見られる。なお、1 段の場合については動作周波数が高くなるため、ローカルクロックを生成するためのリングオシレータの駆動力を高めるために余分に回路規模が必要となる。よって回路規模あたりのスループットで比較すると図 6 のようになり、 $N = 2$ の設計が最も高い値を示しており最もバランスの良い設計であると言える。

また、1 データを処理するのに消費されるエネルギーでは、1, 2, 3 段が小さく、段数が増えるにつれて増加している。これは以下の理由によるものと考えられる。1 つは段数が大きい方が回路規模が増加することによる。もう 1 つは文献 [6] で指摘されているように、段数の多い設計では組み合わせ回路の段数が深くなり、レジスタで消費される電力に比べ組み合わせ回路での電力が支配的となり全体として消費電力が増加する現象が見られることによる。

エネルギーに関しては $N = 3$ が $N = 2$ よりもやや優れているが、本稿では回路規模あたりのスループットが最も優れている $N = 2$ の仮数部除算モジュールを採用することにする。

4. 非同期浮動小数点除算回路の実装及び評価

本章では、3. 章で選択した 2 段の減算シフト回路を持つ仮数部除算モジュールの前後に正規化や丸めを行なう回路を接続し、さらに入出力レジスタとハンドシェイク方式で接続した非同期浮動小数点除算モジュールの FPGA 実装を行う。また、提案浮動小数点モジュールについて既存の同期式回路と比較評価を行う。

4.1 提案浮動小数点除算モジュールの構成

提案する非同期浮動小数点除算モジュールのブロック図を図 7 に示す。これはプロセッサ等の浮動小数点除算が必要な何らかの同期式アーキテクチャに組み込まれたとき、その同期式システムの任意のグローバルクロックに対応でき、しかも最高の

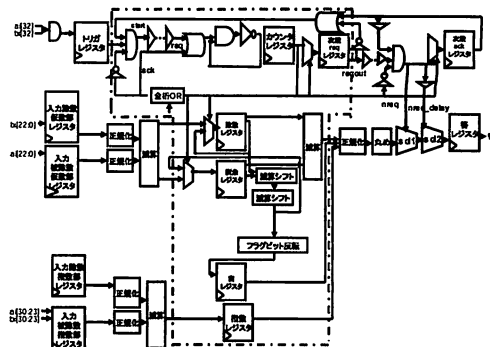


図 7 非同期浮動小数点除算器のブロック図

パフォーマンスが達成可能な浮動小数点除算モジュールである。

図 7 のカウンタレジスタ、除数レジスタ、剰余レジスタ、商レジスタ、指数レジスタは内部のリングオシレータに同期して動作し、入力除数仮数部レジスタ、入力被除数仮数部レジスタ、入力除数指数部レジスタ、入力被除数指数部レジスタおよび次段 ack レジスタ、出力レジスタは外部のグローバルクロックに同期して動作する。点線で囲まれた部分が 3. 章で述べた仮数部除算モジュールであり、その前後の入出力レジスタとはハンドシェイク方式で接続されている。このため仮数部除算モジュールはグローバルクロックによらず最高の速度で動作することができ、仮数部除算モジュールの外にあるレジスタはグローバルクロックで動作しているため他の同期式システムに組み込むことができる。

提案浮動小数点除算モジュールの動作を説明する。まずグローバルクロックに同期して入力除数および被除数がそれぞれレジスタに格納される。仮数部および指数部レジスタに格納された値が減算や正規化の組み合わせ回路を通り、仮数部除算モジュール内のレジスタに到着した後にトリガレジスタの信号 req も遅延回路を通り仮数部除算モジュールに到着し、リングオシレータの動作を開始させる。減算シフト回路を $N = 2$ 段としているため、25 ビットの仮数部を計算するために 13 サイクルかけて減算シフトを繰り返し、除算を行う。

除算が終了すると次のローカルクロックのタイミングでカウンタを 0 に初期化すると同時に仮数部除算モジュールは次段へのリクエスト信号 reqout を立てる。遅延回路を通った reqout 信号である nreq が次段へ到着したことを受けてグローバルクロックのタイミングで出力レジスタは正規化と丸めを行ったデータを出力レジスタに格納する。

4.2 浮動小数点除算モジュールの評価

上述の提案浮動小数点除算モジュールを FPGA に実装し、回路規模、消費電力、スループットの評価を行った。比較の対象としては文献 [6] の同期式浮動小数点除算回路を用いた。ただし、この同期式設計はローム社 $0.35\mu\text{m}$ プロセスで 50, 75, 100, 125, 150 MHz にそれぞれ最適化された 5 種類の ASIC 用ソフト IP コアであるため、FPGA 上で必ずしも最適な設計であるとは限らない。そこでこの設計を元に、FPGA 向けに減算シフト段数を調整し、最適化を行った同期式浮動小数点除算

表 1 仮数部除算モジュールの性能

| 段数 N | 1 | 2 | 3 | 4 | 6 | 8 | 12 | 24 |
|-----------------------------|--------|--------|--------|--------|--------|--------|--------|--------|
| 動作周波数 [MHz] | 160.6 | 91.6 | 68.9 | 50.9 | 34.2 | 27.9 | 18.6 | 8.66 |
| 所要クロックサイクル | 25 | 13 | 9 | 7 | 5 | 4 | 3 | 2 |
| 回路規模 [Slices] | 146 | 129 | 165 | 179 | 229 | 303 | 416 | 839 |
| 消費電力 [mW] | 409 | 446 | 469 | 494 | 571 | 694 | 846 | 1511 |
| 1 データあたりの消費エネルギー [pJ] | 64.7 | 64.3 | 62.3 | 69.0 | 84.8 | 199.9 | 138.1 | 351.0 |
| スループット [MFLOPS] | 6.42 | 7.04 | 7.65 | 7.27 | 6.83 | 6.97 | 6.20 | 4.33 |
| スループット/回路規模 [MFLOPS/Slices] | 0.0440 | 0.0546 | 0.0464 | 0.0428 | 0.0298 | 0.0230 | 0.0149 | 0.0052 |

器を6通り用意した。ターゲットの動作周波数はそれぞれ 25, 50, 75, 100, 115, 130 MHz である。非同期式回路については、同一の回路を上記 6 種類の周波数で動作する同期システムに組み込むことを想定し、外部クロックを 6 種類変化させて評価を行った。

同期式、非同期式の各モジュールの性能を表 2 に示す。以下、回路規模、消費電力、スループットについて同期、非同期の比較を行う。

4.2.1 回路規模

同期式、非同期式の周波数毎の回路規模を図 8 に示す。回路規模については、評価を行った全ての周波数に渡り、非同期式が同期式より小さくなっている。同期式は各周波数で減算シフト段数の異なる設計となっているため、回路規模にばらつきが見られる。それに対し、非同期のものとは同一の設計を 6 種類の外部周波数で動作させているため回路規模は一定である。同期式に比べ、非同期式では回路規模を小さく抑えることができた。

なお、提案非同期回路においては随所に遅延素子を用いている。本研究では FPGA 上への実装を行うため LUT を多段接続することにより遅延素子を実現した。本実装では単純に LUT を直列に接続したため LUT 数が増え、回路規模の増大につながっているが、順序回路を用いる等の工夫により LUT 数を減らすことが可能である。これを行うことにより、全体の回路規模を削減することができ、さらに同期式設計より小さな回路を実現することが可能である。

4.2.2 消費電力

各設計の消費電力を評価するため、1 データを処理するのに要するエネルギーを求めたものを図 9 に示す。

消費エネルギーについても同期式設計には周波数により変化が見られるが、非同期式ではどの周波数においても一定である。同期式がばらつく原因については、1 データあたりのエネルギーとして正規化しているため、スループットの違いによるものである。スループットが周波数により大きく異なる理由については後述する。

25, 50, 130 MHz においては非同期式が同期式より優れた性能を示しているが、その他の周波数ではやや劣っている。先に述べたような回路規模の削減を行うことにより、非同期式設計の消費電力についても削減が可能と考えられる。

4.2.3 スループット

各設計のスループットを図 10 に示す。これまでと同じく、非同期式回路については動作周波数によらずほぼ一定の性能を示

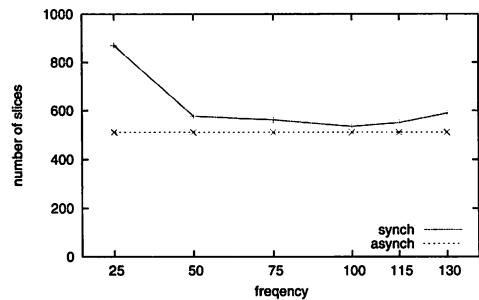


図 8 浮動小数点除算器の回路規模

している。一方、同期式は周波数により性能のばらつきがみられる。これは同期式設計の場合、減算シフトやレジスタなどの取りうる回路構成が限られており演算に要するサイクル数が離散値となることによる。そのため動作周波数とうまく調和すれば高いスループットが得られるが、そうでなければ性能が低下してしまい、設計によりばらつきが生じることになる。

非同期設計は安定したスループットを達成しているが、今回の実装ではほぼ全ての周波数で同期式より低い値となっている。理想的には非同期式回路は外部の動作周波数に影響されないため、その回路が持ちうる最高の性能を発揮できるはずである。しかし本設計において同期式回路よりスループットが低い結果となっているのは、以下の原因が考えられる。今回の設計では 2. 章で述べたように、仮数部除算モジュールを高速なローカルクロックで動作させ、その前後モジュール及び外部の同期回路とはハンドシェイク方式により接続されている。仮数部除算モジュール自体は最高性能を発揮できると考えられるが、その周辺のハンドシェイク通信において待ち時間が発生し、全体としてのスループットの低下を招いていると考えられる。よってこの部分を改良することによりスループットが向上できると考えられる。

スループットと回路規模の関係を評価するため、回路規模あたりのスループットを求めたものを図 11 に示す。25, 50, 130 MHz では非同期式が同期式を上回っているが、その他の周波数では同期式が上回る結果となっている。全体としては非同期式がやや劣る部分があるものの、ほぼ同期式と同程度の安定した性能を示していると言える。非同期式回路においては、上述のような改良を行うことにより回路規模の削減及びスループットの向上を達成すれば、全周波数に渡り同期式より優れた性能を示すことが期待される。

表 2 同期式及び非同期式浮動小数点除算回路の性能

| | 同期式 | | | | | | 非同期式 | | | | | |
|--------------------------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| | 25.0 | 50.0 | 75.0 | 100.0 | 115.0 | 130.0 | 25.0 | 50.0 | 75.0 | 100.0 | 115.0 | 130.0 |
| 動作周波数 [MHz] | 25.0 | 50.0 | 75.0 | 100.0 | 115.0 | 130.0 | 25.0 | 50.0 | 75.0 | 100.0 | 115.0 | 130.0 |
| 回路規模 [Slices] | 870 | 577 | 562 | 535 | 550 | 589 | 511 | 511 | 511 | 511 | 511 | 511 |
| 消費電力 [mW] | 637 | 489 | 469 | 440 | 478 | 431 | 421 | 421 | 421 | 421 | 421 | 421 |
| 1 データあたりの | | | | | | | | | | | | |
| 消費エネルギー [pJ] | 129.4 | 89.5 | 70.0 | 76.5 | 71.9 | 100.9 | 85.6 | 85.7 | 85.6 | 85.7 | 85.6 | 85.6 |
| スループット [MFLOPS] | 5.00 | 5.56 | 6.82 | 5.88 | 6.76 | 4.33 | 5.00 | 5.00 | 5.00 | 5.00 | 5.00 | 5.00 |
| スループット/回路規模 [MFLOPS/Slices] | 0.0057 | 0.0096 | 0.0121 | 0.0110 | 0.0123 | 0.0074 | 0.0098 | 0.0098 | 0.0098 | 0.0098 | 0.0098 | 0.0098 |

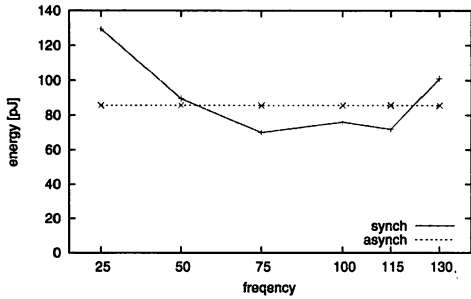


図 9 浮動小数点除算器の消費エネルギー

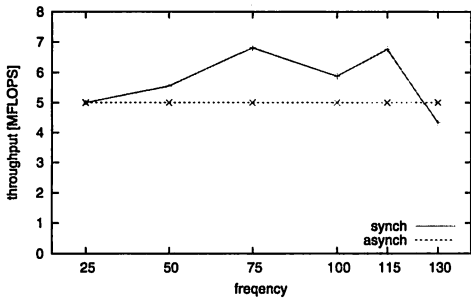


図 10 浮動小数点除算器のスループット

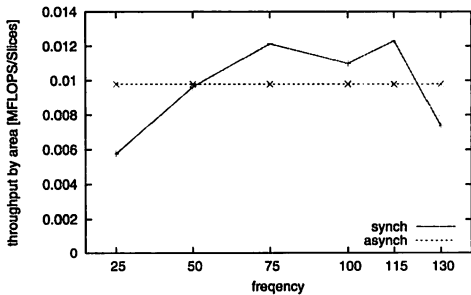


図 11 浮動小数点除算器の回路規模あたりのスループット

5. 結 論

本稿では、非同期式設計を用いることで、単一の IP コアで任意の動作周波数の同期式システムに組み込むことのできる、IEEE-754 準拠の浮動小数点除算モジュールを提案した。提案する浮動小数点除算モジュールは、データの入出力は外部の

同期式システムに同期して行い、残りの仮数部の除算を行うモジュールにはグローバルクロックとは無関係に高速動作するローカルクロック信号を供給し、その前後の同期式モジュールとはハンドシェイク方式で接続したものである。6 種類の周波数に最適化された同期式設計と性能を評価し、回路規模、消費電力についてはどの周波数においても提案除算器が優れていることを示した。また、回路規模、消費電力あたりスループットについても同期式とほぼ同等の値を達成し、動作周波数によらず安定した性能を発揮できることを示した。

今後の課題としては、非同期浮動小数点除算器を改良し、回路規模、消費電力の削減とスループットの向上を行うことが挙げられる。また、今回は論理合成、配置配線及びシミュレーションを行ったが、実際の FPGA に実装し動作確認及び性能の実測を行う予定である。

謝辞 本研究は一部、日本学術振興会科学研究費補助金基盤研究 (C)18500036 による。また、本研究は東京大学大規模集積システム設計教育研究センターを通じ、メンター株式会社の協力で行われたものである。

文 献

- [1] 南谷：“非同期式マイクロプロセッサの動向”，情報処理，Vol39，No.3，pp. 181-186 (1996)。
- [2] T. Sakurai and T. Kuroda: “Tutorial on Low-Power Design Methodology”, Proc. of the Synthesis and System Integration of Mixed Technologies (SASIMI), pp. 3-10 (1996)。
- [3] T. Kawanami, M. Hioki, H. Nagase, T. Tsutsumi, T. Nakagawa, T. Sekigawa and H. Koike: “Preliminary Evaluation of Flex Power FPGA: A Power Reconfigurable Architecture with Fine Granularity”, IEICE Trans. on Inf. and Syst., vol.E87-D, no.8 (2004)。
- [4] K. Katsuki, M. Kotani, K. Kobayashi and H. Onodera: “Extracting a Random Component of Variation from Measurement Results of a 90 nm LUT Array”, Synthesis and System Integration of Mixed Technologies (SASIMI) (2006)。
- [5] D. M. Chapiro: “Globally-Asynchronous Locally-Synchronous Systems”, PhD thesis, Stanford University (1984)。
- [6] H. Ochi, T. Suzuki, S. Matsunaga, Y. Kawano and T. Tsuda: “Development of an IP Library of IEEE-754-Standard Single-Precision Floating-Point Dividers”, IEICE Trans. Fundamentals, vol.E86-A, no.12, pp. 3020-3027 (2003)。
- [7] J. Sparsø and S. B. Furber: “Principles of Asynchronous Circuit Design: A Systems Perspective”, Kluwer Academic (2001)。
- [8] “Ieee standard for binary floating-point arithmetic”, ANSI/IEEE Std754-1985 (1985)。