

令和 2 年 4 月 27 日現在

機関番号：14301

研究種目：基盤研究(B) (一般)

研究期間：2016～2019

課題番号：16H02795

研究課題名(和文) 浮動小数点関数計算のハードウェアアシストに関する研究

研究課題名(英文) Studies on hardware assist of floating point function calculation

研究代表者

高木 直史 (Takagi, Naofumi)

京都大学・情報学研究科・教授

研究者番号：10171422

交付決定額(研究期間全体)：(直接経費) 11,800,000円

研究成果の概要(和文)：指数関数、対数関数、正弦および余弦関数について、高基数STL法あるいは超高基数CORDICと低次の多項式近似およびテーブル参照による丸め補正を組み合わせた、正確丸めを実現する、FPGA実現向けの計算法を開発し、定数テーブルと短冊形乗算器からなるFPGA上での専用計算回路の構成法を示した。逆正弦および逆余弦関数について、新たに高基数CORDIC法を提案し、定数テーブルと短冊形乗算器からなるFPGA上での専用計算回路の構成法を示した。通常の動作中に回路内の単一故障に起因する出力誤りを自ら検出でき、かつ、回路内部の故障を速やかに検出できる加算器の構成法を開発した。

研究成果の学術的意義や社会的意義

科学技術計算において現れる倍精度浮動小数点関数計算を通常倍精度浮動小数点演算器等を用いてソフトウェアで行うには、多大な計算時間とエネルギーを要する。本研究で開発したFPGA実現向けの高精度関数計算手法は、コンピュータに書き換え可能なハードウェアであるFPGAを計算アクセラレータとして付加し、対象となる科学技術計算に応じて必要な関数計算の専用回路をFPGA上に構成することにより、倍精度浮動小数点関数計算を高速にエネルギー効率よく行うことを可能にするもので、コンピュータの高性能化に寄与するものと期待できる。

研究成果の概要(英文)：For exponential function, logarithmic function, and sine / cosine function, we have developed calculation methods suitable for FPGA implementation which realize correct rounding, by combining high-radix STL or ultra-high-radix CORDIC with low-order polynomial approximation and table-based rounding error correction. We have shown construction methods of dedicated circuits consisting of look-up tables and rectangular multipliers on FPGA. For inverse sine / inverse cosine function, we have newly proposed a high-radix CORDIC method, and have shown a construction method of a dedicated circuit consisting of look-up tables and rectangular multipliers on FPGA. We have developed a method of constructing an adder that can detect an output error due to a single failure in the circuit during normal operation and can easily detect a failure in the circuit.

研究分野：計算機工学

キーワード：計算機システム 関数計算 浮動小数点演算 FPGA

## 様式 C - 19、F - 19 - 1、Z - 19 (共通)

### 1. 研究開始当初の背景

2008年に改定された浮動小数点演算標準 IEEE754-2008では、科学技術計算等における計算精度を保証するため、指数関数、対数関数、三角関数等、36個の関数を「正確に丸めるべき関数」として推奨している。正確に丸めるとは、無限精度の関数値をIEEE標準で規定する丸め方法で丸めた値を計算結果とすること、すなわち、定められた精度で正確に計算することである。これを受け、関数計算ライブラリの開発、修正が行われている。これらのライブラリでは、倍精度の正確に丸められた結果を得るために、計算の中間結果を二つの倍精度浮動小数点数の組で表す double-double と呼ばれる表現法等を用いている。double-double の演算では、加算におよそ10回、乗算には20回以上の倍精度浮動小数点演算が必要である。関数計算を通常の倍精度浮動小数点演算器を用いてソフトウェアで行うには、多大な計算時間とエネルギーを要する。

高性能マイクロプロセッサでは、搭載するコア数を増やして並列処理による高性能化が行われているが、処理の並列化の困難さ等から、最近ではSIMD演算機構の強化等により各コアの性能向上が図られている。ゲーム機用プロセッサ等では関数計算ユニットを設けているものもある。また、書き換え可能なハードウェアであるFPGA(Field Programmable Gate Array)を計算アクセラレータとして付加し、実行するプログラムに応じてFPGAを再構成してプログラム中に現れる処理の専用計算回路を構成することが提案されている。FPGAは構成データを書き換えることで論理や配線を自由に変更できるハードウェアであり、年々、ますます大規模、高速で高機能な製品が提供されており、試作品のみならず最終製品にも使われるようになってきている。

FPGAには高速の桁上げ伝搬機構が備わっており、また、DSPスライスと呼ばれる乗算累算器を用いて長ビットの乗算器を構成できる。FPGA上に関数計算回路を構成する研究はいくつかなされているが、対象とする関数が限られており、いずれも限られた範囲の引数に対する24ビット程度までの計算であり、倍精度浮動小数点数を対象とするものは見当たらない。ゲーム機用プロセッサ等に見られる関数計算ユニットは、対象とする関数が限られている。

### 2. 研究の目的

本研究は、IEEE754-2008で「正確に丸めるべき関数」として推奨された36個の関数を高速にエネルギー効率よく計算するためのハードウェアアシスト手法を開発することを目的とする。具体的には、FPGAを再構成して個々の関数の専用計算回路を実現する手法、および、多くの関数の計算に有用な複合演算器を核とする関数計算ユニットを構成する手法の開発を目指す。研究期間内に、以下を明らかにする。

各関数について、ハードウェア実現向きのアルゴリズム(計算手順)を開発し、各関数の専用計算回路のFPGAでの実現法を明らかにする。

各関数について、ハードウェア実現向きのアルゴリズムを探究し、多くの関数の計算に有用な複合演算を明らかにし、その複合演算回路の構成法、および、それを用いた各関数の計算法を明らかにする。

### 3. 研究の方法

三年間で個々の関数について、多項式近似法、収束法等、さまざまな計算法を、計算の並列化、計算時間、中間結果の精度(ビット長)、ハードウェア量等の観点から検討し、FPGA実現に適した専用計算回路の構成法を開発する。精度の解析等は数式処理ソフトウェアを用いて行った。計算時間やハードウェア量の解析等は計算機シミュレーションおよびFPGA評価ボードを用いて行った。

最終年で、ハードウェア実現向きのアルゴリズムをさらに探究し、多くの関数の計算に有用な複合演算を明らかにし、その複合演算回路の構成法を開発するとともに、それを用いた各関数の計算法を開発した。

### 4. 研究成果

#### (1) 指数関数のFPGA実現向き専用計算回路の構成法

指数関数について、高基数法(高基数STL法)と低次の多項式近似(テーラー展開)およびテーブル参照による丸め補正を組み合わせた、正確丸めを実現する、FPGA実現向きの計算法を開発し、FPGA上での専用計算回路の構成法を示した。

任意の数  $P$ 、 $Q(>0)$ 、 $A(>0)$  に対して、 $P+\ln Q=(P-\ln A)+\ln QA$  が成り立つ。 $P_{j+1}=P_j-\ln A_j$ 、 $Q_{j+1}=Q_j A_j$  として、 $P_0=X$ 、 $Q_0=1$  とし、 $P_0+\ln Q_0=P_1+\ln Q_1=\dots=P_n+\ln Q_n$  という変換を考え、 $P_n \rightarrow 0$  となるようにすると、 $Q_n \rightarrow e^X$  となる。入力を固定小数点表現に変換して  $P_0=X$  とし、 $P_0$  の上位数ビットを  $a$  とし、 $P_1=P_0-a$  とし、テーブル参照により  $Q_1=e^a$  を得る。以降、 $P_j$  の上位数ビットを  $a_j$  とし、テーブル参照により  $\ln(1+a_j)$  を得て  $P_{j+1}=P_j-\ln(1+a_j)$ 、 $Q_{j+1}=Q_j+Q_j \times a_j$  を求める。 $\ln(1+a_j)=a_j-a_j^2/2+a_j^3/3-\dots$  であるから  $P_{j+1}$  は0に近づく。 $Q_j \times a_j$  は短冊形乗算(短ビットの数と長ビットの数の乗算)となる。 $P_n \rightarrow 0$  となれば、 $e^X=1+P_n$  あるいは  $1+P_n+P_n^2/2$  により  $e^X$  を求め、 $e^X=Q_n \times e^{\ln}$  とする。この乗算は、 $e^{\ln}$  を数ビットずつに分割することにより、短冊形乗算の繰り返しとなる。

倍精度浮動小数点指数関数計算においては、正確丸めのために、仮数を最大157ビットの精度で計算する必要があることが知られている。正確丸めのために非常に高精度の計算を要する「丸

め困難入力」への対処法として、テーブル参照により丸め補正が必要かどうかを判定することにより、計算の精度を抑えて乗算器等のサイズを小さくする手法を提案した。

これらの計算は、短冊形乗算とテーブル参照によって行える。FPGA では、DSP ブロック内の短ビットの乗算器と高速キャリーチェーンを用いて短冊形乗算器を構成できる。また、LUT あるいは RAM ブロックを用いてテーブルを構成できる。

丸め困難ケースについてさらに解析を行い、仮数を 114 ビット目まで正確に求めれば、補正テーブルを用いずに正確丸めが行えることを明らかにした。これにより、補正テーブルが不要となり、FPGA の必要リソース量が削減できる。

#### (2) 対数関数の FPGA 実現向き専用計算回路の構成法

対数関数について、高基数 (STL) 法と低次多項式近似およびテーブル参照による丸め補正を組み合わせた、正確丸めを実現する、FPGA 実現向きの専用計算回路の構成法を開発した。

$X=2^{X_0}(1+X_f)$  ( $0 < X_f < 1$ ) のとき、 $\ln X = X_0 \ln 2 + \ln(1+X_f)$  である。 $P_{j+1} = P_j - \ln A_j$ 、 $R_{j+1} = (1+R_j)A_j - 1$  とし、 $P_0 = 0$ 、 $R_0 = X_f$  とし、 $P_0 + \ln(1+R_0) = P_1 + \ln(1+R_1) = \dots = P_n + \ln(1+R_n)$  という変換を考え、 $R_n > 0$  となるようにすると、 $P_n \ln(1+X_f)$  となる。入力の仮数部  $X_f$  を  $R_0$  とし、 $R_0$  の上位数ビットを  $b$  とし、テーブル参照により  $1/(1+b)$  の近似値  $A_0$  と  $P_1 = -\ln A_0$  を得て、 $R_1 = (1+R_0)A_0 - 1$  を求める。以降、 $R_j$  の上位数ビットを  $a_j$  とし、テーブル参照により  $\ln(1-a_j)$  を得て  $P_{j+1} = P_j - \ln(1-a_j)$ 、 $R_{j+1} = R_j - a_j - R_j \times a_j$  を求める。 $R_{j+1}$  は 0 に近づく。 $R_j \times a_j$  は短冊形乗算となる。 $R_n > 0$  となれば、 $\ln(1+R_n) = R_n$  あるいは  $R_n - R_n^2/2$  により  $\ln(1+R_n)$  を求め、 $\ln(1+X_f) = P_n + \ln(1+R_n)$  とする。

FPGA 上に短冊形乗算器とテーブルを構成して、専用計算回路を構成できる。

#### (3) 正弦および余弦関数の FPGA 実現向き専用計算回路の構成法

正弦および余弦関数について、超高基数 CORDIC と低次多項式近似およびテーブル参照による丸め誤り補正を組み合わせた、FPGA 実現向きの専用計算回路の構成法を開発した。

CORDIC 法では、平面上のベクトルの回転・伸張の繰り返しにより関数値を求める。 $C_{j+1} = C_j - a_j S_j$ 、 $S_{j+1} = S_j + a_j C_j$  とすると、ベクトル  $(C_{j+1}, S_{j+1})$  は、 $(C_j, S_j)$  を原点を中心に  $\text{atan} a_j$  回転し、 $(1+a_j^2)^{1/2}$  倍伸張したものとなっている。入力  $X$  の上位数ビットを  $a$  とし、テーブル参照により  $C_0 = \cos a$  と  $S_0 = \sin a$  を得、 $X_0 = X - a$ 、 $T_0 = 1$  とする。以降、 $X_j$  の上位数ビットを  $a_j$  とし、 $C_{j+1} = C_j - a_j S_j$ 、 $S_{j+1} = S_j + a_j C_j$  を求めるとともに、テーブル参照により  $\text{atan} a_j$  および  $(1+a_j^2)^{-1/2}$  を得て  $X_{j+1} = X_j - \text{atan} a_j$ 、 $T_{j+1} = T_j (1+a_j^2)^{-1/2}$  を求める。 $\text{atan} a_j = a_j - a_j^3/3 + a_j^5/5 - \dots$  であるから  $X_{j+1}$  は 0 に近づく。 $X_n > 0$  となれば、 $\cos X_n = 1$  あるいは  $1 - X_n^2/2$ 、 $\sin X_n = X_n$  により  $\cos X_n$  と  $\sin X_n$  を求め、 $\cos X = C_n T_n \cos X_n - S_n T_n \sin X_n$ 、 $\sin X = S_n T_n \cos X_n + C_n T_n \sin X_n$  とする。

FPGA 上に短冊形乗算器とテーブルを構成して、専用計算回路を構成できる。

#### (4) 逆正弦および逆余弦関数の FPGA 実現向き専用計算回路の構成法

逆正弦および逆余弦関数について、新たに高基数 CORDIC 法を提案し、FPGA 実現向きの専用計算回路の構成法を開発した。逆正弦および逆余弦計算については、基数 2 の CORDIC 法しか知られていなかった。テーブル参照を削減するために、CORDIC の二重回転法を採用した。

$C_{j+1} = C_j - 2a_j S_j - a_j^2 C_j$ 、 $S_{j+1} = S_j + 2a_j C_j - a_j^2 S_j$  とすると、ベクトル  $(C_{j+1}, S_{j+1})$  は、 $(C_j, S_j)$  を原点を中心に  $2\text{atan} a_j$  回転し、 $1+a_j^2$  倍伸張したものとなっている。逆正弦を求める場合、入力  $X$  の上位数ビットを  $a$  とし、 $S_0 = a$ 、 $T_0 = X$  とし、テーブル参照により  $C_0 = (1-a^2)^{1/2}$  と  $R_0 = a \sin a$  を得る。以降、 $C_j$  の上位ビットから  $C_j$  の逆数の近似値  $\text{inv}_j$  をテーブル参照で得、 $(T_j - S_j) \text{inv}_j / 2$  の上位数ビットを求め  $a_j$  とし、 $C_{j+1} = C_j - 2a_j S_j - a_j^2 C_j$ 、 $S_{j+1} = S_j + 2a_j C_j - a_j^2 S_j$ 、 $T_{j+1} = T_j (1+a_j^2)$  を求めるとともに、テーブル参照により  $\text{atan} a_j$  を得て  $R_{j+1} = R_j + 2\text{atan} a_j$  を求める。 $T_{j+1} - S_{j+1}$  は 0 に近づき、 $R_{j+1}$  は  $a \sin X$  に近づく。

FPGA 上に短冊形乗算器とテーブルを構成して、専用計算回路を構成できる。

#### (5) 関数計算ユニットの構成法

多くの関数がテーブル参照と短冊形乗算の繰り返しにより計算できること示し、関数計算ユニットの核となる演算回路として短冊形乗算器が有用であることを明らかにした。また、計算に必要な定数をいくつかの定数の和で表すことにより、分解した定数を多くの関数計算で共有し、必要なメモリ容量を削減する手法を考案した。

#### (6) 関数計算回路の評価基盤としての自立走行ロボットシステム

関数計算回路を FPGA 上に実現してプロセッサと協調動作させた際の有効性を評価する評価基盤として、自立走行ロボットシステムを開発した。

#### (7) テスト容易な高信頼加算器

最も基本的な演算回路であり、関数計算でも必要な加算器について、通常の動作中に回路内の単一故障に起因する出力誤りを自ら検出でき、かつ、回路内部の故障を速やかに検出できる加算器の構成法を開発した。回路内の単一故障に起因する出力誤りを検出できる加算器においては、出力誤りを生じる入力が増加されない限り故障が生じたことを検出できず、第一の故障を検出できないまま第二の故障が生じると、出力誤りの検出が保証できなくなる。開発した加算器では、10 種の入力が増加されれば単一故障を検出できる。

5. 主な発表論文等

〔雑誌論文〕 計1件（うち査読付論文 1件/うち国際共著 0件/うちオープンアクセス 1件）

1. 著者名 Nobutaka Kito, Naofumi Takagi	4. 巻 68
2. 論文標題 Concurrent Error Detectable Carry Select Adder with Easy Testability	5. 発行年 2019年
3. 雑誌名 IEEE Transactions on Computers	6. 最初と最後の頁 1105-1110
掲載論文のDOI（デジタルオブジェクト識別子） 10.1109/TC.2019.2895074	査読の有無 有
オープンアクセス オープンアクセスとしている（また、その予定である）	国際共著 -

〔学会発表〕 計8件（うち招待講演 0件/うち国際学会 1件）

1. 発表者名 Hideki Takase, Yasuhiro Nitta, So Tamura
2. 発表標題 mROS and ZytteBot: Design Platforms for Embedded Robot Systems
3. 学会等名 Design, Automation and Test in Europe Conference（国際学会）
4. 発表年 2019年

1. 発表者名 藤原康史、高木一義、高木直史
2. 発表標題 高基数S T L法を用いたF P G A 向き対数関数計算法
3. 学会等名 情報処理学会 D A シンポジウム
4. 発表年 2017年

1. 発表者名 藤原康史、高木一義、高木直史
2. 発表標題 高基数S T L法を用いたF P G A 向き指数関数計算法
3. 学会等名 電子情報通信学会 リコンフィギャラブルシステム研究会
4. 発表年 2017年

1. 発表者名 豊島悠紀夫、高木直史
2. 発表標題 倍精度浮動小数点正弦・余弦関数のFPGA向き計算法
3. 学会等名 電子情報通信学会 コンピュータシステム研究会
4. 発表年 2018年

1. 発表者名 藤原康史、高木一義、高木直史
2. 発表標題 倍精度浮動小数点对数関数のFPGA向き計算法
3. 学会等名 電子情報通信学会 コンピュータシステム研究会
4. 発表年 2017年

1. 発表者名 松岡裕志、高木直史
2. 発表標題 逆正弦・逆余弦計算の高基数CORDICアルゴリズム
3. 学会等名 電子情報通信学会 VLSI設計技術研究会
4. 発表年 2019年

1. 発表者名 松岡裕志、高木直史、高木一義
2. 発表標題 逆正弦の高基数CORDICアルゴリズムのFPGA上での実現
3. 学会等名 電子情報通信学会 VLSI設計技術研究会
4. 発表年 2020年

1. 発表者名 高木直史、高木一義
2. 発表標題 IEEE754 binary64における指数関数の丸め困難ケースについて
3. 学会等名 2020年電子情報通信学会総合大会
4. 発表年 2020年

〔図書〕 計0件

〔産業財産権〕

〔その他〕

-

6. 研究組織

	氏名 (ローマ字氏名) (研究者番号)	所属研究機関・部局・職 (機関番号)	備考
研究分担者	高木 一義  (Takagi Kazuyoshi)  (70273844)	三重大学・工学研究科・教授   (14101)	
連携研究者	鬼頭 信貴  (Kito Nobutaka)  (90630997)	中京大学・工学部・講師   (33908)	
連携研究者	高瀬 秀希  (Takase Hideki)  (50633690)	京都大学・情報学研究科・准教授   (14301)	