除算回路のハードウェア設計とその評価

非会員 三 留 浩 幸 (日立製作所)

非会員 ポン マン ヤン (横浜国立大学)

正 員 石 井 六 哉 (横浜国立大学)

Hardware Design of Dividers and It's Evaluation

Hiroyuki Mitome, Non-member (Hitachi, Ltd.)

Pong Mang Yan, Non-member, Rokuya Ishii, Member (Yokohama National University)

Using PARTHENON, a CAD (Computer Aided Design) software system, we design five kinds of fixed point divider, one floating point divider, and synthesize their logic circuits. We compare our designed circuits in point of the numbers of gates, area, power consumption and the maximum of rising delay.

As a results, in case of fixed point division circuits, the gate numbers in each circuits is expressed with their division bits and different variables. The delay time increase depends on the division bits increase, but its increasing rate is difference by different circuits.

The floating point divider was implemented by index calculation, extra process, rounding process to the fixed point divider. The delay time in the floating point divider is longer than that in the fixed point divider only by the amount of time needed to perform the added process.

キーワード： 固定小数点除算器, 浮動小数点除算器

1. まえがき

従来、DSP はチップ単体で製造され、汎用的に使用することが一般的であった。現在では DSP の機能を CAD システムで利用できる形にした DSP コアと呼ばれるものもあり、DSP も特定用途向けに使用されるようになってきている。しかし、DSP や DSP コアには除算専用のハードウェアを持ったものは少なく、高速な除算を必要とするデジタル信号処理を行なうためには除算専用ハードウェアが必要である。

そこで、各種の除算方式を用いて除算回路を設計し、それらの回路規模、遅延速度などの特徴を求めておけば、除算ハードウェアを持った DSP コアや特定用途向け IC を設計する際に、要求される回路規模、遅延速度などの仕様に応じた除算ハードウェアを選択することが可能である。

本論文では NTT データ通信株式会社が開発したハードウェア記述言語 SFL(Structured Function description Language) を中心とした LSI 設計システムパルテノンを用いて行なう。このシステムには回路の設計の他、動作シミュレーション、論理合成など機能を備えており、それらの機能を用いることによって実際のハードウェアの回路規模、動作速度などをかなり正確に求めることができ、評価も容

易に行なえる。ここで、減算シフト型の固定小数点除算回路5種類について設計し、それらの論理合成を行ない、除算方式によってどのように回路の規模、遅延などが変化していくか調べる。さらに、設計した固定小数点除算回路を使用して IEEE754 標準基本単精度浮動小数点形式の除算を行なう浮動小数点除算回路の設計、論理合成も行なう。

第2章では設計する除算器の仕様を述べ、3章で固定小数点除算回路と固定小数点除算器を用いた浮動小数点除算回路の設計を述べ、第4章で設計した除算器の論理合成を行ない、評価を行なう。

2. 除算回路の仕様

<2.1> 固定小数点除算回路の仕様

本論文では n ビットの被除数と n ビットの除数の除算を行ない、 n ビットの商を求める単精度の固定小数点除算回路の設計を行なった ($n = 4, 8, 16, 32$)。以下にそのデータ形式、入出力の構成を示す。

<2.1.1> 固定小数点のデータ形式

本論文で設計した固定小数点除算回路で取り扱う入出力のデータ形式は2の補数表示とし、図1に示すように MSB

(Most Significant Bit) が符号ビット、以下のビットが小数点となる形式とした。

図1 固定小数点のデータ形式

この形式で表現できる実数の範囲は $1 - 2^{-(n-1)}$ から -1 の間で、0を除く絶対値の最小値は $2^{-(n-1)}$ となる。

<2.1.2> 入出力

固定小数点の除算を行なう際に検出できなければならない例外としては、

- 0 除算 (Zero Division : ZD)
 - 無効演算 (Invalid Operation : IO)
 - 固定小数点形式のオーバーフロー (Over Flow : OF)
- の3つがある。ここで、設計した固定小数点除算回路の入出力構成を図2に示す。

図2 固定小数点除算回路の入出力構成

図2に示すように、入力は除数、被除数の $2n$ ビット、出力は商と例外フラグの $n + 3$ ビットである。また、除数、被除数、商は図1に示した形式で表されているものとする。

<2.2> 浮動小数点除算回路の仕様

浮動小数点の扱いはこれまで各種提案、実装されてきたが、現在ではIEEE標準2進浮動小数点演算規格(以下、IEEE754標準規格と記す)で規格化されている(3)-(5)。本論文で設計した浮動小数点除算回路は、このIEEE754標準規格に基づいたものとした。ここではIEEE754標準規格のデータ形式と入出力構成を示す。

<2.2.1> 浮動小数点のデータ形式

IEEE754標準規格では浮動小数点形式を基本形式の単精度、倍精度、拡張形式の単精度、倍精度と4種類規定しているが、本論文ではこの中で必須とされている基本単精度浮動小数点形式を使用した。図3にこの形式を示す。

基本単精度浮動小数点形式は、図3に示すように符号、指数、仮数の3つのフィールドからなる。図3において S は符号ビットであり、0が正、1が負を表す。 F は仮数部であるが、基本形式では $1 \leq F < 2$ の形に正規化されている。このため仮数部のMSBは必ず1となるので、このMSBの1ビットを省略し、MSBより下位の23ビットを F によって表している。 E は8ビットの指数部であり、実際の値に

対してバイアスとして基本単精度の場合は127を加えた値を示している。

図3 IEEE754標準基本単精度浮動小数点形式

正規化された値を式で表現すると次のようになる。

$(-1)^S \cdot 2^{E-127} \cdot (1 + F \times 2^{-23}) \dots\dots\dots (1)$

ただし、 $1 \leq E \leq 254$

<2.2.2> 入出力

IEEE754標準規格では次の5種類の例外が検出できなければならない。

1. 精度 (precision, 結果不正確: inexact result)
2. アンダーフロー
3. 0 除算 (zero divide)
4. オーバーフロー
5. 無効演算

この問題に対して丸めと無限大を用いて修正できる。ここで、入力はIEEE754標準基本単精度浮動小数点形式で表された、除数、被除数の2値、64ビットの他、IEEE754標準規格の丸めモード、無限大モード、例外の演算状態を識別するための8ビットの状態フラグを設けた。出力は商、32ビットと、除算によって生じた例外などの演算状態を反映させた、8ビットの状態フラグの2値、40ビットを出力する。今回設計した浮動小数点除算器の入出力構成を図4に、状態フラグのビット構成を図5に示す。

図4 浮動小数点除算回路の入出力構成

図5 演算状態フラグのビット構成

3. 除算器の設計

<3.1> 固定小数点除算回路の設計

本論文では除算回路によく使われている基本な除算方式の5種類(2進復帰法、2進非復帰法、基数4、8の高基数復帰法、基数8復帰法セル配列)を用い、それぞれ除算ビット数が4、8、16、32ビットの4種類、計20の固定小数点除算回路を設計した。設計した固定小数点除算回路の状態数は図6に示すように、前処理に2つ、実際の除算処理と、商の出力に1ずつの計4つとした。

図6 固定小数点除算回路の状態割り当て

<3.2> 浮動小数点除算回路の設計

浮動小数点除算回路が行なわなければならない処理としては、指数部の計算、仮数部の除算(2進復帰法、基数4、8の高基数復帰法の三つの除算方式を用いた)、除算例外の検出などがあるが、本論文ではそれらを以下に示す4つの処理に分けて考え、除算回路の設計を行なった。

1. 除算前処理部
2. 指数計算部
3. 仮数除算部
4. 除算後処理部

以上述べた処理手順で浮動小数点除算回路を構成した。そのフローチャートを図7に示す。

4. 評価

前章で設計した除算器の動作検証をバルテノンの動作シミュレーター SECONDS で行なった。そして、論理合成ソフトウェア SFLEXP で論理合成を行なう。その後、基本ゲートの伝播遅延、消費電力、面積などの物理情報が記述されている実部品ライブラリを用いて最適化ツール OPT.MAP で実部品マッピングを行ない、クロックなどのタイミング条件を考慮して最適化を行なった。これらのソフトウェアは Sun SPARC Station IPX 上で動作させたが、合成結果が得られるまでに要した時間は4ビットの除算回路で数分、32ビットの基数8セル配列型除算回路で約6時間であった。

<4.1> 固定小数点除算回路の結果と評価

各固定小数点除算回路の論理合成結果を図8、9、10、11に示す。

図7 浮動小数点除算の処理手順フローチャート

これらの結果から、設計した固定小数点除算回路の特徴について、以下のことがわかった。

1. 2進復帰型、2進非復帰型、基数4、8の高基数復帰型除算回路

(a) 各除算回路の回路規模(ゲート数、面積、消費電力)は除算ビット数に比例して増加。

(b) 各除算回路の遅延速度は除算ビット数に対し、ほぼ $\log_2(\text{除算ビット数})$ 程度の増加

(c) 2進復帰型回路、2進非復帰型除算回路の除算実行クロック数は除算ビット数と同程度のクロック数が必要

(d) 基数4復帰型除算回路の除算実行クロック数は除算ビット数の1/2程度のクロック数が必要

(e) 基数8復帰型除算回路の除算実行クロック数は除算ビット数の1/3程度のクロック数が必要

2. 基数8復帰型セル配列型除算回路

(a) 除算ビット数の約2乗に比例して回路規模が増加

(b) 4ビット除算回路以外は、他の同一除算ビットの除算回路に比べ、数倍～約10倍の遅延速度

(c) 除算実行クロック数は除算ビット数によらず4クロック

ここで、簡単のため各除算回路を以下の記号を用いて表す。

m_1 ... 2進復帰型除算回路

m_2 ... 2進非復帰型除算回路

m_3 ... 基数4復帰型除算回路

m_4 ... 基数8復帰型除算回路

図8 各固定小数点除算回路のゲート数

図10 各固定小数点除算回路の消費電力

図9 各固定小数点除算回路の面積

図11 各固定小数点除算回路の最大立ち上がり遅延

$m_5 \cdots$ 基数8復帰型セル配列除算回路

図8の結果から、ゲート数 $G_{m_i}(n) \propto n$ であることがわかった。ただし、基数8復帰型セル配列除算回路の場合では、 $G_{m_5}(n) \propto n(n-1)$ である。更に、除算回路におけるレジスタや前処理部のゲート数は除算回路によらず、一定であるため、各固定小数点除算回路のゲート数 $G_{m_i}(n)$ は除算ビット数 n を用いて、以下のように表すことが出来ると考えられる。

$$G_{m_i}(n) \simeq \frac{1}{4}(an + b_{m_i}n) \cdots \cdots (2)$$

ただし、 $i=1,2,3,4$

基数8復帰型セル配列除算回路の場合は除算実行部を構成回路の個数はほぼ除算ビット数の1/3倍ずつ増加するので、以下のように表すことが出来ると考えられる。

$$G_{m_5}(n) \simeq \frac{1}{4}\left(an + \frac{b_{m_5}}{3}n(n-1)\right) \cdots \cdots (3)$$

ここに、 a : 4ビット除算器におけるレジスタや前

処理部のゲート数、 b_{m_i} : 各4ビット除算器における除算実行部のゲート数。

表1に示す a 、 b_{m_i} は本論文で使用した実部品ライブラリにシステムパルテノンに付属してきたアモ用1.5μCMOSテクノロジーのライブラリを使用したもので、実際に設計を行なうとき、式(2)、(3)を用いて、パラメータ a 、 b_{m_i} の値を求めれば、他の異なる実部品ライブラリや機能回路にはそのまま適応できる。 a 、 b_{m_i} の値を求めることで設計前に除算回路の大まかなゲート数を知ることが可能となる。

表1 各除算回路の a と b_{m_i} の値

a	b_{m_1}	b_{m_2}	b_{m_3}	b_{m_4}	b_{m_5}
180	212	132	372	796	540

ゲート数 G_{m_i} と面積 A_{m_i} と消費電力 P_{m_i} の相互関係式は以下のように表すことができる。

$$G_{m_i} \simeq 1.20A_{m_i} \dots\dots\dots(4)$$

$$G_{m_1} \simeq 0.29P_{m_1}$$

$$G_{m_2} \simeq 0.30P_{m_2}$$

$$G_{m_3} \simeq 0.27P_{m_3}$$

$$G_{m_4} \simeq 0.26P_{m_4}$$

$$G_{m_5} \simeq 0.27P_{m_5} \dots\dots\dots(5)$$

式(4)を考えると、ゲート数と面積は除算回路によらずに一定な比率で増加することがわかった。式(4)と式(5)から面積と消費電力の関係式を求めることができる。

図11から、ゲート数 n と最大立ち上がり遅延 $T_{m_i}(n)$ の関係式は以下のように表すことができる。

$$T_{m_1}(n) \simeq 27.6 \log_2 n + 11.8$$

$$T_{m_2}(n) \simeq 34.0 \log_2 n + 3.0$$

$$T_{m_3}(n) \simeq 21.0 \log_2 n + 59.7$$

$$T_{m_4}(n) \simeq 26.6 \log_2 n + 42.5$$

$$T_{m_5}(n) \simeq 7.9(\log_2 n)^3 \dots\dots\dots(6)$$

<4.2> 浮動小数点除算回路の結果と評価

浮動小数点除算回路は固定小数点除算回路と同じ方法で動作シミュレーションを行ない、論理合成・最適化を行なった。その結果を図12、13、14、15に示す。

図12 各浮動小数点除算回路のゲート数

図12、13、14に示した浮動小数点除算回路の回路規模(ゲート数、面積、消費電力)は仮数の除算に使用した復帰型固定小数点除算回路の基数に対して同様の増加傾向を示している。そして、浮動小数点除算回路全体のゲート数から仮数の除算に使用する固定小数点除算回路のゲート数を引いた残りの部分のゲート数は約1500ゲートと一定となっている。

以上のことから、本論文で設計した浮動小数点除算回路は32ビットの固定小数点除算回路に指数の計算回路、例外処理回路、丸め回路など約1500ゲートを追加することで実現できるということがわかった。

図13 各浮動小数点除算回路の面積

図14 各浮動小数点除算回路の消費電力

図15 各浮動小数点除算回路の最大立ち上がり遅延

ゲート数 G と面積 A と消費電力 P の相互関係式は以下のように表すことができる。

$G \simeq 1.20A \dots\dots\dots (7)$

$G \simeq 0.24P \dots\dots\dots (8)$

式(4)と式(7)を考えると、除算回路においてゲート数は面積の1.2倍であることがわかった。

図15では最大立ち上がり遅延は基数に比例していないので、定式化のは困難である。ところで、遅延速度については各浮動小数点除算回路のクリティカル・パスを調べてみた所、仮数の固定小数点除算回路の部分であった。図11に示した32ビット固定小数点除算回路の遅延速度の値と浮動小数点除算回路の遅延速度の値を以下の表に示す。

表2 浮動小数点除算回路と固定小数点除算回路の遅延時間

	浮動小数点除算回路	固定小数点除算回路
2進復帰型	171.9	164.9
基数4復帰型	194.6	160.7
基数8復帰型	191.2	173.5

単位:[ns]

表2より、浮動小数点除算回路の遅延速度と固定小数点除算回路の遅延速度はほぼ同程度の値となっているが、浮動小数点除算回路の遅延速度の方が若干大きな値となっている。これは、浮動小数点除算回路は仮数の除算を行なう前に例外処理を行なうこと、また除算後に丸めやデノーマル処理も行なうことから、同ビット数の固定小数点除算回路に比べ遅延速度が大きくなると考えられる。

5. むすび

本論文では、減算シフト型の固定小数点除算回路5種類について除算ビット数が4、8、16、32ビットのものを設計、論理合成し、除算方式、除算ビット数によって回路の規模、遅延などがどの様に変化していくか調べた。また、設計した固定小数点除算回路を使用してIEEE754標準基本単精度浮動小数点形式の除算を行なう浮動小数点除算回路の設計、論理合成も行なった。

その結果、固定小数点除算回路については、除算回路の回路規模、遅延速度が求まり、回路規模を除算ビット数を変数に用いて表すことが出来た。遅延時間については、除算ビット数が増えると共に遅延時間も増す傾向にあるが、その増加の度合は除算回路の構成方法により異なるということがわかった。

浮動小数点除算回路については、IEEE754規格に沿ったものを設計することが出来、浮動小数点除算の例外処理部などの回路規模を求めることが出来た。

今後の研究課題としては、本論文で設計した除算回路を使用したデジタル信号処理回路の設計などが上げられる。

(平成7年10月31日)

文 献

(1) NTT データ通信株式会社：“PARTHENON Reference Manual”
(2) NTT データ通信株式会社：“PARTHENON User's Manual”
(3) 富澤 治：“VLSI 用コンピュータアーキテクチャ”，昭晃堂，1989
(4) “Proposed Standard for Binary Floating-Point Arithmetic Draft 8.0 of IEEE Task P754”，IEEE Computer, pp.51-62, March 1981
(5) 小林 史典，“IEEE 浮動小数点演算規格の概要と背景”，インターフェイス, pp.218-230, March 1985
(6) Kai Hwang 著 堀越 他 訳：“コンピュータの高速演算方式”，近代科学社，1980
(7) 三留 浩幸, 守田 知紀, 石井 六哉, “パルテノンによる除算器の設計”，電気学会研究会資料, 電子回路研究会, ECT-93-41, pp.23-29, September 1993
(8) 三留 浩幸, 守田 知紀, 石井 六哉, “パルテノンによる除算器の設計”，第8回デジタル信号処理シンポジウム講演論文集, pp.1-6, October 1993

三留 浩幸 (非会員) 92年3月横浜国立大学電子情報工学科卒業。94年3月同大学大学院修士課程修了。同年日立(株)入社。デジタル信号処理に関する研究に従事。

ボンマン ヤン (非会員) 1971年3月8日生まれ。94年3月横浜国立大学電子情報工学科卒業。現在、同大学大学院修士課程在学中。デジタル信号処理に関する研究に従事。

石井 六哉 (正員) 昭和42年東工大・工・電子卒・昭和47年同大学大学院博士過程了。同年日立製作所中央研究所入所。昭和50年横浜国大・工・電子情報助教授。現在、教授。工博。回路理論、デジタル信号処理に関する研究に従事。電気学会、計測自動制御学会、IEEE各会員。