選択型セル配列高速除算器の設計とVLSI評価

非会員 恒川 佳隆 (岩手大)

非会員 日野杉充希 (岩手大)

正 員 三浦 守 (岩手大)

Design and VLSI Evaluation of High-Speed Cellular Array Divider with Selection Function
Yoshitaka Tsunekawa, Non-member, Mitsuki Hinosugi, Non-member, Mamoru Miura, Member (Iwate University)

In recent years, a very high-speed divider is required in real-time applications of digital signal processing and robot control and so on. In this paper, a high-speed cellular array divider with selection function is proposed, which is based on the non-restoring algorithm and can deal with both fixed-point and negative operands in two's complement form. This divider uses new techniques which can generate in parallel both a quotient bit of one row and a partial remainder and CLA bit of its next row. Moreover the delay time of the proposed divider is calculated in terms of a delay of one unit such as NAND gate. Finally, by using PARTHENON, a CAD (Computer Aided Design) system for VLSI, this divider is designed and evaluated. As a result, elimination of delay time for even rows becomes possible. Thus, the delay time can decrease by approximately one-half of the high-speed divider proposed by Cappa and Hamacher which uses the most general high-speed techniques of carry-save and CLA.

キーワード：高速除算器，セル配列，非復帰法，VLSI評価

1. まえがき

加減乗除等の算術演算は，計算機などのデジタルシステムにおいて，基本演算として広く用いられており，従来からその高速化が望まれている．算術演算については，既にさまざまな研究がなされており，いろいろなアルゴリズムが提案され，実現されてきた．特に，集積回路の進歩に伴い，LSI化に適したセル配列構造の高速算術演算回路に関する研究が盛んに行われてきた．本論文では，VLSI化に適したセル配列構造の高速除算器について報告する．

これまで，加減算器，乗算器などについては様々な実用化の報告がなされてきている．それに対して，除算器はあまり積極的に実用化がなされてこなかった．その理由として，除算器は普通，他の演算器に比べて使用頻度が余り高くないことがいえる．また，他の演算器と違い，除算は並列的に演算を行うことができないために分散処理ができず，任意の精度に拡張が容易に行えない点などが挙げられる⁽¹⁾．その結果，従来において除算器の使用は，高価格化と低効率化につながってきた．

しかし現在，デジタル信号処理やロボット制御などのリアルタイム性を必要とする分野において，より高速な

除算器が必要とされている．また，近年のASIC(Application Specific IC; 特定用途向けIC)の普及により，ハードウェアの低価格化が進み，大規模で高性能な専用演算器の実現が容易になってきている．さらに，近年の集積回路技術の進歩により，除算器を組合せ回路としてLSI上を実現することが容易となっており，高速でLSI化に適した除算器の開発が望まれている．

除算については既にさまざまな研究がなされており，いろいろな除算法が提案されている⁽²⁾．これらの除算法は，減算シフト型除算法と乗算型(収束型)除算法に大別される．

本論文では，規則正しいセル配列構造でVLSI化に適した減算シフト型除算法を対象とする．また，減算シフト型除算アルゴリズムとして，復帰法(回復型)，非復帰法(非回復型)，SRT法などがある．その中でも非復帰法は被除数，除数共に正規化を必要としないため固定小数点のまま演算ができ，2の補数表現された負の値を直接扱うことが可能である．このことから，デジタル信号処理やロボット制御などの分野での実用化を考え，本論文では非復帰法を用いる．

これまでも減算シフト型除算法としてはいくつかのセル配列除算器が提案されてきている⁽²⁾⁻⁽⁶⁾．しかし，どの

除算アルゴリズムを用いた除算器も前行で求めた部分剰余を必要とするため、演算は1行ずつ逐次的に行われていく。ゆえに、遅延時間は語長に比例して増加し、これが減算シフト型の除算器を高速化する上で大きな問題点となる。

本論文では、被除数、除数が負の場合でも演算可能であり、また、前行の商ビットの生成と次行の部分剰余、CLA信号の生成を並列的に行うという新たな手法を用いたセル配列高速除算器を提案する⁽⁷⁾。さらに、単位ゲート遅延を用いて本除算器の遅延時間を算出する。そして最後に、NTTデータ通信株式会社が開発したVLSI設計システムPARTHENONを用いて本除算器のVLSI評価を行う。

2. 非復帰法アルゴリズム

本章では非復帰法アルゴリズムを説明する。なお、語長を L ビットとする。全ての値が符号ビットを持つので、被除数、剰余は $L+1$ ビット、除数、部分剰余、商は $L/2+1$ ビットである。よって、

$$\begin{aligned} \text{被除数} : N &= n_0, n_1, n_2, \dots, n_L \\ \text{除数} : D &= d_0, d_1, d_2, \dots, d_{L/2} \\ \text{部分剰余} : P &= p_0, p_1, p_2, \dots, p_{L/2} \\ \text{商} : Q &= q_0, q_1, q_2, \dots, q_{L/2} \\ \text{剰余} : R &= r_0, r_1, r_2, \dots, r_L \end{aligned}$$

とする。なお、添え字の0の付いたものは符号ビットである。

以下に、アルゴリズムの手順を示す。

ステップ1

$$\begin{aligned} q_{-1} &:= \begin{cases} 1 & \text{if } N, D \text{ が同符号} \\ 0 & \text{if } N, D \text{ が異符号} \end{cases} \\ P &:= \begin{cases} N - D & \text{if } q_{-1} = 1 \\ N + D & \text{if } q_{-1} = 0 \end{cases} \end{aligned}$$

ステップ2

```
for  $i=0$  step 1 until  $L/2-1$  do
  begin
     $P := P \text{ shl } 1;$ 
     $q_i := \begin{cases} 1 & \text{if } P, D \text{ が同符号} \\ 0 & \text{if } P, D \text{ が異符号} \end{cases}$ 
     $P := \begin{cases} P - D & \text{if } q_i = 1 \\ P + D & \text{if } q_i = 0 \end{cases}$ 
  end
```

ステップ3

$$\begin{aligned} Q &= q_0, q_1, q_2, \dots, q_{L/2-1} \\ R &= 2^{-L/2} \cdot P \end{aligned}$$

3. 従来の非復帰法によるセル配列除算器

これまでに、非復帰法によるセル配列除算器がいくつか提案されている⁽³⁾⁻⁽⁵⁾。本論文では、従来のセル配列高速除算器としてCappaとHamacherによって提案された構成法を基に考える⁽⁵⁾。これはキャリーセーブ方式とCLA回路を組み合わせた除算器で、これらは現在最も一般的に用いられている高速化手法といえる。そこで、本章ではこの除算器の構成法について述べる。ただし、先に提案された除算器では正の値しか扱えないため、ここではさらに負の値まで拡張した構成法について考える。

<3.1> 高速化の手法

この除算器では、CLAとキャリーセーブの手法を用いて高速な除算器を実現している。

除算器を桁上げ伝播加算器で実現する場合、各演算でキャリーが完全に伝播することを必要とするため、遅延時間は非常に大きなものとなる。しかし、この除算器では部分剰余を和とキャリーのベクトルに分けることにより、得られたキャリーベクトルをその行で伝播せずに次行の演算で加算している。これが、キャリーセーブ方式である。

また、部分剰余の符号ビットを高速に生成するために、CLA回路を用いて符号ビットへのキャリーを生成している。

これらの手法を用いたこの除算器の遅延時間は、 $O(L \log L)$ の値となる。

<3.2> 構成法

前述の手法を用いた除算器の語長が8ビットのときの全体の構成を図1(a)に示す。この除算器は、3種類のセルから構成される。

Aセルでは、和 S_j^i とキャリー C_j^i のベクトルに分解した部分剰余を、全加算器によって生成する(図1(b))。さらに、CLA信号生成のための生成信号 G_j^i や伝播信号 P_j^i もここで生成する。

Sセルは各行に1つずつ用いており、仮の符号ビットを生成する(図1(c))。そして、この信号とCLA信号をXOR(排他的論理和)をとって符号ビットを生成し、この信号が次行のSセルに入力され商ビット Q_i を生成する。

CLAセルは各行に1つずつ用いており、各Aセルで生成された生成信号 G や伝播信号 P がこのセルに入力され、符号ビット生成に必要なCLA信号 CL を生成する(図1(d))。

4. 選択型セル配列高速除算器

<4.1> 新たな高速化の手法

2章のアルゴリズムで説明したように、前述の除算器は、各行で部分剰余と除数の符号を比較して商ビットを決め、この商ビットによって次行の演算を決定するため、演算は1行ずつ逐次的に行われていく。これは、非復帰法に限らず、復帰法、SRT法など前行の部分剰余を必要とする除算

(a) Modified Cappa-Hamacher's array divider with 8-bit dividend and 4-bit divisor.

(b) A cell.

(c) S cell.

(d) CLA cell.

図1 改良型Cappa-Hamacherの配列除算器の構成
Fig. 1 Structure of modified Cappa-Hamacher's array divider.

法は全て同様である。これが、一般的に他の演算器と比べて除算器が遅いといわれる最大の理由である。

しかし、非復帰法において各行の制御信号となる商ビットはただか0と1しかない。よって、前行の商ビットが決定する前に商ビットが0と1の場合の演算を行うことにより先行演算が可能となる。そして、商ビットが決定してから演算結果を選択する。これが、新たな手法の基本的な考え方である。

なお、この手法を用いた本除算器を、演算結果を商ビットによって選択することから、選択型セル配列高速除算器と呼ぶことにする。

本手法の具体的手順を以下に示す。

● 部分剰余生成時間の削減

まず本手法の第1段として、前行の商ビットが決定する前に、商ビットが0と1の場合の次行の部分剰余を先に生成するように考える。これによって、加減算の遅延時間を削減することができる。この考え方を図2に示す。

この実現法として、部分剰余を生成するセルに加算用と減算用の2つの全加算器を用いる。また前述の除算器では、図1(a)に示すように、最右端のセルのキャリー入力に前行の商ビットを入力するようになっていた。しかし、本手法で同様にした場合、そのセルだけが商ビットが決定し

Modified Cappa-Hamacher's Divider

Proposed Divider

図2 部分剰余の生成時間の削減

Fig. 2 Elimination of generating time for partial remainder.

ないかぎり演算を行うことができない。そこで、新たに最右端用のセルとしてRセルという新しいセルを用いる。

● CLA信号生成時間の削減

前述の実現法によって、部分剰余の生成時間を削減することができた。しかし、さらに検討してみた結果、CLA信号の生成時間が最大遅延パスとなることがわかった。そこでCLA信号も、商ビットが0と1の場合を先行演算するように考える。よって、CLA信号の生成時間を削減することが可能となる。

この実現法として、まず部分剰余を生成するセルでCLA信号生成に必要な生成信号、伝播信号をそれぞれ商ビットが0と1の場合を生成する。そして、各行に商ビットが0と1の場合のCLA信号生成を行うCLAセルを2つ用いる。

<4.2> 構成法

本除算器の語長が8ビットのときの全体の構成を図3(a)に示す。本除算器の構成は4種類のセルで構成する。

Bセルでは、商ビットが0の場合の部分剰余(和 SO_j^i , キャリー CO_j^i)と1の場合の部分剰余(和 SI_j^i , キャリー CI_j^i)を2つの全加算器によって生成する(図3(b))。そして、商ビットが決定した時点で部分剰余をセレクタによって選択する。また、CLA信号生成のための生成信号や伝播信号も商ビットが0(生成信号 GO_j^i , 伝播信号 PO_j^i)と1(生成信号 GI_j^i , 伝播信号 PI_j^i)の場合をAND, ORゲートを2つずつ用いて生成する。

Rセルは各行の最右端に用いるセルである(図3(c))。動作はBセルとほぼ同じで、最下位ビットの部分剰余の生成を目的としている。しかし、この部分にRセルという新しいセルを用いた理由は、4.1節で述べたようにこの部分にB

セルを用いてキャリー入力に商ビットを入力するよう設計すると、このセルが最大遅延パスとなり本除算器の利点が活かされない。よって、全加算器のキャリー入力に加算演算用には'0'を、また減算演算用には'1'の定数を入力することによって、この問題を解決している。

5. 遅延時間

<5.1> 演算プロセス

基本的に、従来の除算器と本除算器の各行の演算プロセスはほとんど同じであるが、本除算器の全体の演算プロセスは、図4に示すように従来の除算器と大きく異なる。以下に本除算器の演算プロセスを説明するが、そのために図4の1行目、2行目、3行目に注目していただきたい。

まず1行目の演算は、初回の制御信号の生成を待たずに開始している。これは、本除算器では制御信号が0と1の場合の部分剰余を先に演算できるからである。そして1行目の演算は順次に行われていくが、その間にも2行目の演算がすでに1行目の部分剰余の生成が終わった時点で開始している。これは、初回の制御信号がすでに決定しているの、2つのパターンの部分剰余を制御信号によって選択するのみとなっているためである。そして2行目の演算も順に行われていく。さらに、その間にも1行目の商ビットが生成された時点で3行目の演算が開始している。これもすでに2つのパターンの部分剰余は2行目の演算ですでに生成されており、1行目の商ビットによって選択するのみとなっているためである。

よって、この図から容易に2行目、4行目など偶数行の遅延時間が削減できることがわかる。このことから、従来の除算器の遅延時間は語長に比例する値となるが、本除算器では語長の半分に比例する値となる。

図3 選択型セル配列高速除算器の構成
Fig. 3 Structure of high-speed cellular array divider with selection function.

<5.2> 単位ゲート遅延を用いた遅延時間の算出

次に、本除算器の遅延時間を単位ゲート遅延を用いて算出する。なお単位ゲート遅延 Δ とは、論理回路1段あたりの遅延時間に相当し、NANDゲートやNORゲート、NOTゲートを1とする値である⁽²⁾。

前節でも説明したように、基本的に本除算器と従来の除算器の各行の演算プロセスは同じであり、部分剰余の生成、CLA信号の生成、商ビットの生成という順に行われていく。それぞれの生成時間は、部分剰余は 7Δ 、CLA信号は語長が10ビット未満のとき 6Δ 、10ビット以上のとき 8Δ （これは最大ファンイン数を8とした場合、CLA回路を2段にしなければならないことによる⁽⁸⁾）、また、商ビットは 10Δ となることからそれぞれの除算器の1行当たりの遅延時間は、

$$\Delta_{row(従来)} = 21\Delta + \Delta_{CLA} \quad \dots\dots\dots (1)$$

$$\Delta_{row(本除算器)} = 23\Delta + \Delta_{CLA} \quad \dots\dots\dots (2)$$

となる。なお、本除算器と従来の除算器との差が 2Δ あるのは、本除算器ではQセルで商ビットが0と1の場合のCLA信号、部分剰余の最上位ビットの和とキャリーを選択するためである。

よって、各除算器は $L/2$ 行あるので、

$$\Delta_{従来} = (21\Delta + \Delta_{CLA})L/2 + 8\Delta \quad \dots\dots\dots (3)$$

$$\Delta_{本除算器} = (23\Delta + \Delta_{CLA})L/4 - 2\Delta \quad \dots\dots\dots (4)$$

となる。なお、従来の除算器では $L/2$ かけているのに対し

本除算器で $L/4$ かけているのは、前節で説明したように、本除算器は偶数行の遅延時間が削減されるためである。

よって、(3)、(4)式から容易に、

$$\Delta_{本除算器} \approx \Delta_{従来} / 2 \quad \dots\dots\dots (5)$$

となることがわかる。

6. PARTHENONによるVLSI評価

ここではさらに、VLSI設計システムPARTHENONにより本除算器を設計し、VLSI評価を行った。

<6.1> VLSI設計

本除算器を設計するために、本除算器をSFL言語で記述した。PARTHENONではトップダウン設計が可能であるので、各セルをサブモジュールとして扱うことにより容易に設計することが可能となる。また、本除算器を任意の語長に設計できるように、本除算器のSFLソースを自動作成するプログラムをC言語によって実現した。

<6.2> VLSI評価

SFL言語で記述した本除算器を動作シミュレータSECONDSによって動作確認し、さらにシンセサイザSFLEXPで論理合成し、最適化ユーティリティOPT_MAPによってテクノロジー・マッピングと回路の最適化を行った^{(9),(10)}。さらに、本除算器を語長を6ビットから32ビットまで変えてVLSI評価した。評価対象を消費電力、面積、ゲート数、商の生成時間、剰余の生成時間とし、結果を図5に示す。なお、用いた設計ルールは $0.8\mu\text{mCMOS}$ スタン

図4 本除算器の遅延時間
Fig. 4 Delay time of proposed divider.

(a) Power.

(b) Area.

(c) Gates.

(d) Delay time for quotient.

(e) Delay time for remainder.

A... Modified Cappa - Hamcher's divider
 B... Proposed divider
 (elimination of generating time
 for partial remainder)
 C... Proposed divider
 (elimination of generating time
 for both partial remainder and CLA bit)

図5 VLSI評価

Fig. 5 VLSI evaluation.

ダードセル(VLSIテクノロジー社)である。

図5(d)の商の生成の遅延時間からみても、本除算器は従来の除算器より2倍以上高速になったことがわかる。具体的に、語長が24ビットのときで従来の除算器よりも2.35倍の高速化を図ることができた。しかも、消費電力、ゲート数ともに1.75倍、面積は1.81倍と、いずれも2倍以下となる。

また、部分剰余生成時間の削減だけを行った除算器では、1.54倍しか高速化が得られないが、消費電力は1.54倍、

面積は1.50倍、ゲート数は1.42倍となる。

7. むすび

本論文においては、前行の商ビットの生成と次行の部分剰余、CLA信号の生成を並列に行うという新たな手法を用いた選択型セル配列高速除算器を提案した。さらに、本除算器の遅延時間を単位ゲート遅延を用いて算出し、VLSI設計システムPARTHENONによって本除算器を設計

しVLSI評価を行った。

その結果、部分剰余とCLA信号生成時間の削減を図った場合には、偶数行目の遅延時間の削減が可能となり、従来提案されているものに対して約2倍の高速化が可能となることが明らかになった。また、PARTHENONの評価結果より消費電力、ゲート数ともに2倍以下となることが示された。

今後の研究課題としては、本除算器のレイアウト設計での評価を行う必要がある。

謝辞 本研究にあたり、PARTHENONをご提供いただいたNTTデータ通信(株)に深く感謝いたします。

(平成8年9月25日受付)

文献

- (1) 中村次男, 笠原宏: 「任意精度の演算が可能な並列除算器のアーキテクチャ」, 電子情報通信学会論文誌C-II, Vol. J75-C-II No. 5, 185~192 (平4-5)
- (2) Hwang, K.: "Computer Arithmetic/Principles, Architecture, and Design", John Wiley & Sons (1979)
- (3) Dean, K. J.: "Binary Division Using a Data Dependent Iterative Arrays", *Electronics Letters*, Vol. 4, 283~284 (1968-7)
- (4) Guild, H. H.: "Some Cellular Logic Arrays for Nonrestoring Binary Division", *The Radio and Elec. Engr.*, Vol. 39, 345~348 (1970-6)
- (5) Cappa, M. and Hamacher, V. C.: "An Augmented Iterative Array for High-Speed Binary Division", *IEEE Trans. on Computers*, Vol. C-22, 172~175 (1973-2)
- (6) 高木直史, 安浦寛人, 矢島修三: 「冗長2進表現を利用したVLSI向き高速除算器」, 電子通信学会論文誌, Vol. J67-D No. 4, 450~457 (昭59-2)
- (7) 日野杉充希, 恒川佳隆, 吉田等明, 三浦守: 「選択型セルアレイ高速除算器の設計とVLSI評価」, 計測自動制御学会東北支部 第160回研究集会資料, 160-14 (平8-5)
- (8) 久津輪敏郎: 「並列乗除算配列におけるCLA回路の効果」, 電子通信学会論文誌, Vol. J64-D No. 8, 797~798 (昭56-8)
- (9) NTTデータ通信株式会社: "PARTHENON Reference Manual" (1990)
- (10) NTTデータ通信株式会社: "PARTHENON User's Manual" (1990)

恒川 佳隆

(非会員)昭55岩手大・工・電気卒。昭58東北大学大学院修士課程了。同年岩手大・工・助手。平6同大講師。工博。デジタル信号処理, デジタル制御の研究に従事。IEEE, 計測自動制御学会会員。

日野杉 充希

(非会員)平8岩手大・工・情報卒。現在, 同年同大学大学院工学研究科博士前期課程在学中。高速除算器の構成に関する研究に従事。

三浦 守

(正員)昭37岩手大・工・電気卒。同年東北大・工・助手。昭43秋田工業高専助教授。昭47岩手大・工・助教授。昭60同大教授。工博。この間, マグネティックス, デジタル信号処理などに関する研究に従事。平1年10月より10カ月間、ウインザ大(カナダ)およびカリフォルニア大学サンタバーバラ校(米国)の客員研究員。計測自動制御学会, IEEE, 情報処理学会, 日本ME学会各会員。