

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第6064435号
(P6064435)

(45) 発行日 平成29年1月25日(2017.1.25)

(24) 登録日 平成29年1月6日(2017.1.6)

(51) Int.Cl.

G06F 7/535 (2006.01)

F 1

G06F 7/535

請求項の数 7 (全 19 頁)

(21) 出願番号 特願2012-182344 (P2012-182344)
 (22) 出願日 平成24年8月21日 (2012.8.21)
 (65) 公開番号 特開2014-41415 (P2014-41415A)
 (43) 公開日 平成26年3月6日 (2014.3.6)
 審査請求日 平成27年5月12日 (2015.5.12)

(73) 特許権者 000005223
 富士通株式会社
 神奈川県川崎市中原区上小田中4丁目1番
 1号
 (74) 代理人 100107766
 弁理士 伊東 忠重
 (74) 代理人 100070150
 弁理士 伊東 忠彦
 (74) 代理人 100146776
 弁理士 山口 昭則
 (72) 発明者 篠宮 研介
 神奈川県川崎市中原区上小田中4丁目1番
 1号 富士通株式会社内

最終頁に続く

(54) 【発明の名称】演算回路、演算処理装置、及び除算方法

(57) 【特許請求の範囲】

【請求項 1】

引き戻し法による除算を行う演算回路であって、
 中間剰余を格納する中間剰余レジスタと、
 前記中間剰余の上位2桁と除数の上位1桁との情報に基づいて、前記情報から予測可能な精度よりも低い精度での商予測を行うことにより、予測結果を生成する商予測回路と、
 前記商予測回路により生成された前記予測結果に応じて選択されたN倍 (Nは自然数) の除数を出力する定数倍回路と、

前記定数倍回路が出力する前記N倍の除数を前記中間剰余から減算する減算器と、
 前記減算器による減算のキャリーアウトに応じて部分商を求める部分商計算回路と
 を含むことを特徴とする演算回路。

【請求項 2】

前記商予測回路が前記情報に基づいて生成する前記予測結果は、前記部分商の取り得る全ての値を含む集合の部分集合であるグループを示す信号であり、互いに重複する複数のグループのうちの何れか1つのグループを特定する信号であることを特徴とする請求項1記載の演算回路。

【請求項 3】

前記互いに重複する複数のグループの数は、2つのグループであることを特徴とする請求項2記載の演算回路。

【請求項 4】

10

20

前記商予測回路は、組み合わせ論理回路であることを特徴とする請求項 1 乃至 3 何れか一項記載の演算回路。

【請求項 5】

前記定数倍回路は前記予測結果に応じて選択された複数の N 倍の除数を出力し、前記減算器は前記複数の N 倍の除数をそれぞれ入力とする複数の減算器を含むことを特徴とする請求項 1 乃至 4 何れか一項記載の演算回路。

【請求項 6】

引き戻し法による除算を行う演算回路と、前記除算の命令をデコードする命令制御部とを有する演算処理装置において、

前記演算回路は、

10

中間剰余を格納する中間剰余レジスタと、

前記中間剰余の上位 2 枠と除数の上位 1 枠との情報に基づいて、前記情報から予測可能な精度よりも低い精度での商予測を行うことにより、予測結果を生成する商予測回路と、

前記商予測回路により生成された前記予測結果に応じて選択された N 倍（N は自然数）の除数を出力する定数倍回路と、

前記定数倍回路が出力する前記 N 倍の除数を前記中間剰余から減算する減算器と、

前記減算器による減算のキャリーアウトに応じて部分商を求める部分商計算回路とを含むことを特徴とする演算処理装置。

【請求項 7】

引き戻し法による除算を行う演算回路において、

20

中間剰余の上位 2 枠と除数の上位 1 枠との情報に基づいて、前記情報から予測可能な精度よりも低い精度での商予測を行うことにより、予測結果を生成し、

前記予測結果に応じて選択された N 倍（N は自然数）の除数を前記中間剰余から減算し、

前記減算のキャリーアウトに応じて部分商を求める

各段階を含むことを特徴とする除算方法。

【発明の詳細な説明】

【技術分野】

【0001】

本願開示は、演算回路、演算処理装置、及び除算方法に関する。

30

【背景技術】

【0002】

2 進化 10 進数（B C D : Binary Coded Decimal）の四則演算において、除算は他の演算と比較して演算サイクル数を必要とする低速な演算である。一般に精度の高い除算は引き戻し法によって部分商と中間剰余とを求めており、その中間剰余の生成がクリティカルな要因となる。基本的な引き戻し法では、中間剰余から除数を減算する処理を繰り返す。減算結果が負になった時点で、減算しそぎであると判断し、そのサイクルの減算前の結果を部分商として用いる。

【0003】

以下に、引き戻し法の手順を説明する。なお以降の説明において、断りの無い限り被除数と中間剰余とを区別せず、中間剰余と表記する。まず中間剰余レジスタ、除数レジスタ、及び部分商レジスタから、中間剰余、除数、及び部分商が提供される。減算ループの初回において、部分商は 0 である。初回又はその後の減算ループにおいて、以下の処理が実行される。まず部分商をカウントアップする。次に減算回路により中間剰余から除数を減算し、減算の結果とキャリーアウトとを生成する。キャリーアウトが 1 であるならば（即ち結果が正の数）、減算結果を中間剰余レジスタに格納し、現在の減算ループの最初にカウントアップされた部分商を部分商レジスタに格納し、次の減算ループに移行する。キャリーアウトが 0 であるならば（即ち結果が負の数）、中間剰余レジスタの値（現在の減算ループの減算前の値）を中間剰余レジスタに格納し、部分商レジスタの値（現在の減算ループのカウントアップ前の値）を部分商レジスタに格納し、処理を止める。この時の中間

40

50

剰余レジスタと部分商レジスタとの値が、最終的な中間剰余と部分商との結果となる。

【0004】

このように引き戻し法は、部分商と中間剰余の生成の為に、中間剰余が負になるまで中間剰余から除数を減算する処理を繰り返す必要がある。10進数の場合、1桁の商は0～9の範囲を取りえる為、減算処理を最大で10回繰り返す必要がある。これを全桁の商が導出されるまで繰り返す必要があるため、除算を実現する演算器のレイテンシは非常に低くなってしまう。

【0005】

基本的な引き戻し法の問題は、部分商と中間剰余とを生成する減算ループの繰り返し回数が多いことである。この問題を解決する為に、除数のN倍数（Nは自然数）をいくつか10
予め計算しておき、中間剰余からそれぞれのN倍された除数を減算して場合分けするという手法が一般にとられる。

【0006】

例えば除数の1, 2, 5倍数を予め計算しておく手法がある（例えば特許文献1）。一回目の減算で中間剰余から5倍の除数を減算し、結果が負の数である時、除数の5倍は減算しすぎであるとわかる。その為、1桁の商は0～4の範囲であることが判明する。それ以外の時は1桁の商が5～9の範囲であることが判明する。このようにN倍の除数を用意して粗く引き戻し法を行うことにより次のサイクルでの商の取りえる範囲を減らし、部分商と中間剰余の生成ループ回数を減らすことが可能である。特許文献1では、このアルゴリズムによって最大4回のループ回数で結果を得られるとしている。
20

【0007】

また特許文献1では1個の減算器を利用しているが、複数個の減算器を用意して同時に複数個の除数のN倍数を減算した結果を用意すれば、さらなる高速化が可能となることは既知である。極端な例を挙げれば、1～9倍の除数を予め用意しておき、9個の減算器を使用すれば、1回のループで結果を得ることができる。また例えば1, 2, 3, 6倍の除数を予め用意しておき、更に2つの減算器を用いて結果を得る回路が考えられる（例えば特許文献2）。

【0008】

先のN倍除数の減算による高速化と合わせて、被除数と除数の状態から部分商と中間剰余を予測するという手法も存在する。例えば2回目の減算をする際に中間剰余と3倍除数の上位桁の状態を見ることにより、2回目の減算に使うN倍除数を選択する回路等が考えられる（例えば特許文献2）。また中間剰余と除数の状態から誤差が1以下の精度で部分商を予測できる商予測回路と、その誤差の補正回路を追加して高速化を図ること等が考えられる（例えば特許文献3）。

【0009】

複数個のN倍除数を用意することによる高速化においては、回路規模の増大とループ回数の低減とがトレードオフの関係にある。従ってハードウェアの制約により少ない減算器を使っての除算が必要である場合、結果を得るために必要なサイクル数は多くなるを得ない。また、商予測による高速化においては追加回路の増大がボトルネックとなる。部分商及び部分剰余生成のループ内に制御回路を組み込むとなると、ループ数低減によるレイテンシの向上が果たせたとしても、ループ内の論理段数が重くなり、高い動作周波数での実装が困難となる。
40

【0010】

また商予測及び商補正が高速であっても、剰余の種類が多かったり、3Nの固定数を乗算する演算回路を用いたりする場合には問題がある（例えば特許文献2及び3）。10進演算器において、3Nの固定数を乗算する演算回路は加算器を使わなければ実現できない。これを実現するには以下の3つの方法が考えられる。

(1) 加算器の直前に加算器を追加する。

(2) 減算器と共有する。

(3) 6倍除数をループ前に生成し、レジスタで保持し続ける。
50

このうち(1)の手段をとる場合、論理段数が加算器分増える為、ディレイに悪影響を与える。(2)の手段をとる場合、部分商及び部分剰余の生成の為に1サイクル増やす必要があり、さらには制御が煩雑になる。(3)の手段をとる場合、除数の幅を有するレジスタが追加されるため、面積的に問題となる。

【0011】

また商予測の為に重い論理が必要となる為、高い動作周波数の場合には1サイクル内で商予測と減算とを同時に行なうことが困難になる。その場合、処理サイクルを分けなければならなくなり、レイテンシの悪化する虞れがある。

【0012】

また特許文献2の場合、商予測の手段として中間剰余と3倍除数の上位2桁を比較している。比較回路は、一般には加算器で実現するので、2桁の加算器がさらに必要となってしまう。また高動作周波数の場合には、レイテンシの悪化する虞れがある。

【先行技術文献】

【特許文献】

【0013】

【特許文献1】特開昭57-125442号公報

【特許文献2】特開平07-239774号公報

【特許文献3】特開平07-160480号公報

【発明の概要】

【発明が解決しようとする課題】

【0014】

以上を鑑みると、引き戻し法における減算ループ回数を、効率的な回路構成により削減することが可能な演算回路が望まれる。

【課題を解決するための手段】

【0015】

引き戻し法による除算を行う演算回路は、中間剰余を格納する中間剰余レジスタと、前記中間剰余の上位2桁と除数の上位1桁との情報に基づいて、前記情報から予測可能な精度よりも低い精度での商予測を行うことにより、予測結果を生成する商予測回路と、前記商予測回路により生成された前記予測結果に応じて選択されたN倍(Nは自然数)の除数を出力する定数倍回路と、前記定数倍回路が出力する前記N倍の除数を前記中間剰余から減算する減算器と、前記減算器による減算のキャリーアウトに応じて部分商を求める部分商計算回路とを含むことを特徴とする。

【発明の効果】

【0016】

少なくとも1つの実施例によれば、引き戻し法における減算ループ回数を、効率的な回路構成により削減することが可能な演算回路が提供される。

【図面の簡単な説明】

【0017】

【図1】加算器の数と減算時に商が取りえる数との組み合わせに対し、それぞれ残りの減算回数を示した表である。

【図2】中間剰余の上位2桁と除数の上位1桁との組み合わせテーブルを示す図である。

【図3】2個の加算器により2回のループで結果を算出可能な演算器において、1回目及び2回目の減算の結果の組み合わせに対して結果として得られる部分商の値の一例を示した表である。

【図4】図3に示したアルゴリズムの処理の流れを示すフローチャートである。

【図5】コンピュータシステムの構成の一例を示す図である。

【図6】演算回路の構成の一例を示す図である。

【図7】2倍回路における各桁の入出力値を示す真理値表である。

【図8】5倍回路における各桁の入出力値を示す真理値表である。

【図9】商予測回路の構成の一例を示す図である。

10

20

30

40

50

【図10】定数倍回路の構成の一例を示す図である。

【図11】倍数選択回路の構成の一例を示す図である。

【図12】倍数選択回路及び定数倍回路の入出力の関係を示す表である。

【図13】中間剩余選択回路の構成の一例を示す図である。

【図14】中間剩余選択回路の入出力の関係を示す表である。

【図15】制御回路の構成の一例を示す図である。

【図16】部分商計算回路の構成の一例を示す図である。

【図17】部分商計算回路の入出力の関係を示す表である。

【図18】定数テーブルの入出力の関係を示す表である。

【発明を実施するための形態】

10

【0018】

以下に、本発明の実施例を添付の図面を用いて詳細に説明する。

【0019】

N倍除数を使って除算を行う際、加算器の数によって必要な減算ループ回数は変動する。最適なN倍除数を用いるとすると、必要な減算回数は以下の式で求められる。

$$\log_{a+1} A = B \quad (\text{但し } B \text{ の小数点以下は切り上げ})$$

ここで、aは加算器の数、Aは部分商の取りえる範囲、Bは部分商が得られるまでに必要な最大の減算の回数である。

【0020】

20

図1は、加算器の数と減算時に商が取りえる数との組み合わせに対し、それぞれ残りの減算回数を示した表である。例えば、10進除算の場合、商が0～9の範囲の値を取り得る、即ち商が10通りの値をとりえる為、減算時に商が取りえる数は10となる。図1において、「減算時に商が取りえる数」が10である列を見ると、例えば「減算器の数」が1であれば、残りの減算回数が4であることが分かる。即ち、最適なN倍除数を用いるとすると（例えば最初に5倍の除数を用いるとすると）、残りの減算回数が4であるから、最大4回のループ回数で結果を得ることができる。また「減算時に商が取りえる数」が10である場合において、例えば「減算器の数」が2であれば、残りの減算回数即ちループ回数が3であることが分かる。

【0021】

30

例えば加算器が1つであることを前提とすると、商予測等の何らかの前処理により初期状態での商候補の数を8個に減らすことができれば、商候補の数が10個である場合のループ回数4から、ループ回数を3に減らすことができる。また例えば加算器が2つであることを前提とすると、図1の表から分かるように、商予測等の何らかの前処理により初期状態での商候補の数を9個に減らすことができれば、商候補の数が10個である場合のループ回数3から、ループ回数を2に減らすことができる。

【0022】

商の取り得る数を10個からm個（mは10より小さい自然数）に減らすためには、例えば10個の商候補を、各々がm個以下の商を含む複数のグループに分け、商予測により1つのグループを特定すればよい。ここで各グループは、商が取り得る全ての値（10個の値）を含む集合の部分集合であるグループである。例えば商の取り得る数を10個から9個に減らすためには、10個の商候補を、各々が9個以下の商を含む2つのグループに分け、商予測により1つのグループを特定すればよい。この時、2つのグループは互いに重複してよい。即ち、2つのグループには、同一の商が含まれてよい。

40

【0023】

これはすなわち、特許文献3のように精度の高い商予測を行わずとも、粗い商予測をすることで処理の高速化が充分に見込めることを表している。粗い商予測として、中間剩余の上位2桁と除数の上位1桁との情報に基づいて商予測を考えることを考える。

【0024】

図2は、中間剩余の上位2桁と除数の上位1桁との組み合わせテーブルを示す図である。最も左側の列は被除数（中間剩余）の上位2桁を示し、最も上側の行は除数の上位1桁

50

を示し、行と列とが交差する各欄には、当該被除数と除数との組み合わせに対して部分商が取りうる範囲を示している。この例において、被除数と除数とは、共に10進数である。例えば被除数が $08.\times\times$ ($\times\times$ は任意の数)であり、除数が $01.\times\times$ である時、部分商の取りうる範囲は、図2の表において $4\sim8$ ($4, 5, 6, 7, 8$)となっている。このことは、「 8.00 被除数 <9.00 」と「 1.00 除数 <2.00 」の時、「 4 部分商 8 」であることを示している。

【0025】

図2の表において、網掛けの部分は、引き戻し法のアルゴリズム上、ありえない組み合わせである。この表は 100 行 \times 9 列という膨大なデータ量を含んでいるが、この表に含まれる全てのデータを商予測に組み込む必要はない。例えば、被除数が $08.\times\times$ であり、除数が $01.\times\times$ であることが情報として分かっているとき、この情報から予測可能な精度で予測できる部分商の取りうる範囲は $4\sim8$ である。このとき、この範囲に含まれる商の数は5個である。しかしながら前述のように、例えば加算器が1つであることを前提とすると、商予測等の何らかの前処理により初期状態での商候補の数を8個に減らすことができれば、商候補の数が10個である場合のループ回数4から、ループ回数を3に減らすことができる。また例えば加算器が2つであることを前提とすると、図1の表から分かるように、商予測等の何らかの前処理により初期状態での商候補の数を9個に減らすことができれば、商候補の数が10個である場合のループ回数3から、ループ回数を2に減らすことができる。

【0026】

例えば加算器が2つの場合を考えると、商予測によって商の取りえる数を9個以下にすればよいので、図2の表の全てのデータを商予測に用いる必要はない。各グループに含まれる商の数が9個以下となればよいので、上述の被除数が $08.\times\times$ であり除数が $01.\times\times$ であるという情報から得られる5個の商からなるグループまで特定する必要はない。即ち、被除数(中間剩余)の上位2桁と除数の上位1桁との情報に基づいて、当該情報から予測可能な精度(例えば $4\sim8$ を特定できる精度)よりも低い精度での商予測を行うことにより、商の取りえる数を9個以下にすれば十分である。

【0027】

低い精度での商予測即ち粗い商予測を行うためには、使用する加算器の数及び所望の減算ループの回数に応じて、都合の良い箇所で図2の表を区切ればよい。例えば、2個の加算器が設けられ2回のループで結果を算出可能な演算器を設計するためには、図2の表を区切り線10で示したように区切り、商の取りえる範囲を商が $0\sim7$ であるグループと商が $4\sim9$ であるグループとに分ければよい。これら2つのグループの各々は、部分商の取り得る全ての値($0\sim9$)を含む集合の部分集合であるグループである。またこれら2つのグループは、互いに重複するグループであり、同一の要素 $4, 5, 6, 7$ を共有する。

【0028】

このように各グループに含まれる商の数が9個以下である2つのグループに分け、後述する商予測により、これら複数のグループのうちの何れか1つのグループを特定することができれば、初期状態での商候補の数を9個に減らすことができる。これにより、加算器が2つである場合、図1の表から分かるように、商候補の数が10個である場合のループ回数3から、ループ回数を2に減らすことができる。

【0029】

なお図2の表において、区切り線10で示されるような2つのグループに区切った理由は、商予測を単純な論理で実現できることができることが挙げられる。例えば除数が $01.\times\times$ の列に着目した時、中間剩余が 8 (1000_2)以上とそれ未満(0111_2 以下)とで分けると、商が $0\sim7$ のグループと商が $4\sim9$ のグループとに分けることができ、中間剩余のMSBをチェックするだけで商予測をすることができる。即ち、中間剩余のMSBをチェックするだけで、取り得る商が属するグループとして、2つのグループのうちの何れか1つのグループを特定することができる。例えば仮に商が $0\sim8$ のグループと商が $4\sim9$ のグループとに分けるとすると、中間剩余が 9 (1001_2)以上及びそれ未満となるように

10

20

30

40

50

区切る必要があり、中間剩余の4ビット全てをチェックする必要がある。区切り線10は、中間剩余の全ビットをチェックせずに、成る可く少ない数の一部のビットをチェックするだけでグループ分けできるような分割を実現している。

【0030】

なお上記の区切り線10によるグループ分けの例では、各グループの要素の数（各グループに含まれる商の数）が8個以下となっている。従って、このグループ分けにより、加算器が1つである場合も、図1の表から分かるように、商候補の数が10個である場合のループ回数4から、ループ回数を3に減らすことができる。同様に、図1の表において例えば減算ループ回数を2から1に減らすように、例えば加算器が3個である場合であれば、各グループに含まれる要素の数が4以下となるように、図2の表において3分割するグループ分けを行ってもよい。10

【0031】

本願開示の粗い商予測は、被除数（中間剩余）の上位2桁と除数の上位1桁との情報に基づいて、当該情報から予測可能な精度よりも低い精度での商予測を行うものであり、特定の加算器の数やループ回数に限定されるものではない。この粗い商予測においては、例えば図2の表において、被除数の上位2桁と除数の上位1桁とを特定することにより特定される1つの行と列との交差部分に記載された商の範囲を特定するのではなく、行と列との交差部分を複数個纏めたグループを特定している。またこの粗い商予測においては、例えば被除数（中間剩余）の上位2桁と除数の上位1桁との全てのビットのうち、一部のビットのみを用いて商予測を行ってよい。20

【0032】

図3は、2個の加算器により2回のループで結果を算出可能な演算器において、1回目及び2回目の減算の結果の組み合わせに対して結果として得られる部分商の値の一例を示した表である。この例では、上述のように、商の取りえる範囲を商が0～7であるグループと商が4～9であるグループとに分けるように商予測を行うことを前提としている。この表に示す内容は、一例として想定したアルゴリズムの動作に基づくものであり、何れのN倍除数を用いるのか、どのように部分商や中間剩余を求めるのか等について限定を意図するものではない。

【0033】

図3においては、中間剩余をR、除数をD I V s、部分商をQとしている。また第1及び第2の加算器（減算器）が求めた減算結果（中間剩余）をそれぞれR1及びR2、第1及び第2の加算器（減算器）による減算処理のキャリーアウトをそれぞれC01及びC02としている。図3に示されるように、商の取り得る範囲が例えば4～9であると予測された場合、1回目の減算ループにおいて、2つの減算器による2つの減算として、「R - 5 D I V s」（中間剩余 - 5倍除数）及び「R - 8 D I V s」（中間剩余 - 8倍除数）を実行する。その結果得られるキャリーアウトがそれぞれ例えば「正」と「負」である場合、取り得る商の範囲は5～7であることが分かり、部分商Qとして取り敢えず5を設定し、中間剩余Rとして第1の減算器の減算結果R1を設定する。更に2回目の減算ループにおいて、2つの減算器による2つの減算として、「R - 1 D I V s」（中間剩余 - 1倍除数）及び「R - 2 D I V s」（中間剩余 - 2倍除数）を実行する。その結果得られるキャリーアウトがそれぞれ例えば「正」と「正」である場合、取り得る商は7であることが分かるので、部分商Qを+2して7（=5+2）に設定し、中間剩余Rとして第2の減算器の減算結果R2を設定する。なお図3において「-」や「*」は想定したアルゴリズムでは起こり得ない組み合わせであることを示している。3040

【0034】

図4は、図3に示したアルゴリズムの処理の流れを示すフローチャートである。以下に、このアルゴリズムの処理について説明する。

【0035】

ステップS1で、中間剩余Rと除数D I V sとが入力される。ステップS2で商予測を行う。この商予測は、被除数（中間剩余）の上位2桁と除数の上位1桁との情報に基づい50

て、図2の表で区切り線10の上側のグループ（商が0～7）と区切り線10の下側のグループ（商が4～9）の何れか1つのグループを特定することにより行われる。

【0036】

ステップS3で、取りうる部分商の範囲が4～9であればステップS4の処理を、取りうる部分商の範囲が0～7であればステップS9の処理を行うようにセレクト信号を生成する。

【0037】

ステップS4で、第1及び第2の減算器に中間剰余と除数が入力される。第1の減算器により、中間剰余Rから5倍の除数が減算され、中間剰余R1とキャリーアウトCO1とが出力される。また第2の減算器により、中間剰余Rから8倍の除数が減算され、中間剰余R2とキャリーアウトCO2とが出力される。
10

【0038】

ステップS5で、第1及び第2の減算器のキャリーアウトCO1及びCO2の組み合せから、中間剰余Rと部分商Qとに入力する値を選択する。CO1及びCO2が0及び0であるならば、ステップS6で、中間剰余Rの値をそのままとし、部分商Qを4とする。CO1及びCO2が1及び0であるならば、ステップS7で、中間剰余R1を中間剰余Rとし、部分商Qを5とする。CO1及びCO2が1及び1であるならば、ステップS8で、中間剰余R2を中間剰余Rとし、部分商Qを8とする。

【0039】

ステップS9で、第1の減算器と第2の減算器に中間剰余と除数が入力される。第1の減算器により、中間剰余Rから2倍の除数が減算され、中間剰余R1とキャリーアウトCO1とが出力される。また第2の減算器により、中間剰余Rから5倍の除数が減算され、中間剰余R2とキャリーアウトCO2とが出力される。
20

【0040】

ステップS10で、第1及び第2の減算器のキャリーアウトCO1及びCO2の組み合せから、中間剰余Rと部分商Qとに入力する値を選択する。CO1及びCO2が0及び0であるならば、ステップS11で、中間剰余Rの値をそのままとし、部分商Qを0とする。CO1及びCO2が1及び0であるならば、ステップS12で、中間剰余R1を中間剰余Rとし、部分商Qを2とする。CO1及びCO2が1及び1であるならば、ステップS13で、中間剰余R2を中間剰余Rとし、部分商Qを5とする。
30

【0041】

以上が1回目の減算ループであり、その後、以下に説明する2回目の減算ループが実行される。

【0042】

商予測が4～9の範囲であり且つ1回目の減算ループにおけるCO1及びCO2が0及び0であったならば、ステップS14で、中間剰余Rから4倍の除数が減算され、減算結果が中間剰余Rとなる。

【0043】

商予測が4～9の範囲であり且つ1回目の減算ループにおけるCO1及びCO2が0及び0であったという条件以外の場合には、ステップS15で、第1の減算器と第2の減算器に中間剰余と除数が入力される。第1の減算器により、中間剰余Rから除数が減算され、中間剰余R1とキャリーアウトCO1とが出力される。また第2の減算器により、中間剰余Rから2倍の除数が減算され、中間剰余R2とキャリーアウトR2とが出力される。
40

【0044】

ステップS16で、第1及び第2の減算器のキャリーアウトCO1及びCO2の組み合せから、中間剰余Rと部分商Qとに入力する値を選択する。CO1及びCO2が0及び0であるならば、ステップS17で、中間剰余Rの値をそのままとし、部分商Qもそのままとする。CO1及びCO2が1及び0であるならば、ステップS18で、中間剰余R1を中間剰余Rとし、部分商Qに1を加算する。CO1及びCO2が1及び1であるならば、ステップS19で、中間剰余R2を中間剰余Rとし、部分商Qに2を加算する。
50

【0045】

最後にステップ S 2 0 で、中間剩余 R と部分商 Q とを出力する。以上の処理により、2 回の減算ループで中間剩余 R と部分商 Q とを求めることができる。

【0046】

図 5 は、コンピュータシステムの構成の一例を示す図である。図 5 に示すコンピュータシステムは、プロセッサ 110 及びメモリ 111 を含む。演算処理装置としてのプロセッサ 110 は、2 次キャッシュ部 112、1 次キャッシュ部 113、制御部 114、及び演算部 115 を含む。1 次キャッシュ部 113 は、命令キャッシュ 113A 及びデータキャッシュ 113B を含む。演算部 115 は、レジスタ 116、演算制御部 117、及び演算器 118 を含む。演算器 118 には除算器 119 が含まれる。除算器 119 には、部分剩余及び部分商を求める演算回路 119A が含まれる。なお図 5 及び以降の同様の図において、各ボックスで示される各機能ブロックと他の機能ブロックとの境界は、基本的には機能的な境界を示すものであり、物理的な位置の分離、電気的な信号の分離、制御論理的な分離等に対応するとは限らない。各機能ブロックは、他のブロックと物理的にある程度分離された 1 つのハードウェアモジュールであってもよいし、或いは他のブロックと物理的に一体となったハードウェアモジュール中の 1 つの機能を示したものであってもよい。各機能ブロックは、他のブロックと論理的にある程度分離された 1 つのモジュールであってもよいし、或いは他のブロックと論理的に一体となったモジュール中の 1 つの機能を示したものであってもよい。

【0047】

上記コンピュータシステムは CPU (Central Processing Unit) を用いた情報処理装置を模式化したものであり、このコンピュータシステムにより Oracle-number 等を演算するハードウェアを実現する。プロセッサ 110 では、1 次キャッシュ部 113 及び 2 次キャッシュ部 112 を設けることにより、キャッシュメモリを多階層化した構成となっている。具体的には、1 次キャッシュ部 113 と主記憶 (メモリ 111) との間に、主記憶よりも高速にアクセスできる 2 次キャッシュ部 112 を設けている。これにより、1 次キャッシュ部 113 においてキャッシュミスが発生した場合に、主記憶にアクセスが必要になる頻度を低くして、キャッシュミス・ペナルティーを軽減することができる。

【0048】

制御部 (命令制御部) 114 は、命令フェッチアドレスと命令フェッチリクエストとを 1 次命令キャッシュ 113A に発行し、この命令フェッチアドレスから命令をフェッチする。制御部 114 は、フェッチした命令 (例えば除算命令) をデコードした結果に従い演算部 115 を制御して、フェッチされた命令を実行する。演算制御部 117 は、制御部 114 の制御下で動作し、演算対象のレジスタ 116 からのデータを演算器 118 に供給したり、演算結果のデータを指定されたレジスタ 116 に格納したりする。また演算制御部 117 は、演算器 118 が実行する演算のタイプを指定する。更に演算制御部 117 は、アクセス先のアドレスを指定し、1 次キャッシュ部 113 の当該アドレスに対してロード命令やストア命令を実行する。ロード命令により、指定アドレスから読み出されたデータは、指定されたレジスタ 116 に格納される。またストア命令により、指定されたレジスタ 116 のデータが、指定されたアドレスに書き込まれる。演算器 118 に含まれる除算器 119 の演算回路 119A が、部分商及び中間剩余を計算する回路であり、例えば前述の粗い商予測に基づいて 2 個の加算器により 2 回のループで結果を算出可能な演算回路であってよい。また以下の説明において、

図 6 は、演算回路 119A の構成の一例を示す図である。図 6 に示す演算回路 119A は、中間剩余レジスタ 121、除数レジスタ 122、サイクルレジスタ 123、4 倍選択レジスタ 124、商予測回路 125、倍数選択回路 126、定数倍回路 127、減算器 128、減算器 129、及び制御回路 130 を含む。演算回路 119A は更に、部分商計算回路 131、中間剩余選択回路 132、部分商レジスタ 133、及びセレクタ 134 を含む。

10

20

30

40

50

【0049】

定数倍回路 127 は、2 倍の除数、4 倍の除数、5 倍の除数、及び 8 倍の除数を生成する。例えば 2 進化 10 進数の定数倍を生成するうち、これら 4 つの N 倍除数 (N = 2, 4, 5, 8) は、それ以外の倍数よりも単純な論理で実現することが可能である。

【0050】

2 倍回路においては、各桁を 2 倍すると、キャリー伝搬を除いた各桁が必ず偶数になるので、下位からのキャリー伝搬がそのまま各桁の LSB (最下位ビット) に収まる。従ってキャリーが伝搬していく状況を考える必要がなく、1 つの着目桁の値を計算する際には、当該桁の値とその 1 つ下の桁の値だけを用いればよい。そのようにして 2 倍値を計算する回路は、入力値と出力値とを示す真理値表に基づいて、組み合わせ論理回路として実現することができる。そのようにして実現した回路は、加算器を用いて 2 倍値を計算するよりも高速に 2 倍値を求めることができる。10

【0051】

図 7 は、2 倍回路における各桁の入出力値を示す真理値表である。A_n [3 : 0] は n 桁目の入力である 4 ビットの値である。S_n 及び S_{n+1} は A_n を 2 倍して得られる値を表わし、S_n [3 : 1] は n 桁目の 4 ビットのうちの上位 3 ビットであり、S_{n+1} [0] は n+1 桁目の 4 ビットの最下位ビットである。例えば A_n [3 : 0] が 1000 (10 進で 8) の場合、その 2 倍の値 (10 進で 16) は、n+1 桁が 0001、n 桁が 0110 となる。従って、図 7 に示されるように、S_{n+1} [0] = 1、S_n [3 : 1] = 011 となる。この入力値と出力値とを示す真理値表を実現する組み合わせ論理回路を設計することで、2 倍回路を実現することができる。20

【0052】

4 倍回路及び 8 倍回路の場合は、キャリーが 2 ビット分発生することがある為、上記のように 1 桁の真理値表に基づいて回路を設計することはできない。しかしながら、2 倍回路が単純な組み合わせ論理回路で実現できるので、その 2 倍回路を直列に 2 個繋げることにより 4 倍回路を実現し、また、2 倍回路を直列に 3 個繋げることにより 8 倍回路を実現すればよい。

【0053】

5 倍回路の場合は、入力数を 10 倍して得られた結果を 2 で割る処理を行えばよい。その処理は以下のようにして実現できる。まず入力数を 10 倍処理に相当する 4 ビット左シフトする。得られた入力数 10 倍の値を、1 / 2 倍処理に相当する 1 ビット右シフトする。但し、この 1 ビット右シフト処理において、同一桁内で値が 1 のビットが移動する場合には正しい結果 (1 / 2 倍の結果) が得られるが、n+1 桁から n 桁に値 1 のビットが移動した場合、n+1 桁から n 桁に落ちてきた値は 8 (1000₂) となってしまう。本来は n+1 桁の 1 の 1 / 2 の値は n 桁の 5 に等しいので、n+1 桁から n 桁に落ちてきた値 8 を 5 に変換する必要がある。そこで、各桁において MSB が 1 であった場合、MSB を 0 にして、更にその桁に 5 を加算する。なお 1 / 2 倍処理に相当する 1 ビット右シフトをした時点で、各桁の下位 3 ビットの値は 0 ~ 4 の値しか取り得ないので、上記のようにして 5 を加算してもキャリーアウトは発生しない。そのようにして 5 倍値を計算する回路は、入力値と出力値とを示す真理値表に基づいて、組み合わせ論理回路として実現することができる。そのようにして実現した回路は、加算器を用いて 5 倍値を計算するよりも高速に 5 倍値を求めることができる。3040

【0054】

図 8 は、5 倍回路における各桁の入出力値を示す真理値表である。A_n [0] は n 桁目の入力 4 ビットの最下位ビットであり、A_{n-1} [3 : 1] は n-1 桁目の入力 4 ビットの上位 3 ビットである。この入力を 3 ビット左シフト (= 4 ビット左シフト + 1 ビット右シフト) し、MSB が 1 であった場合、MSB を 0 にして更に 5 を加算して得られる n 桁の出力 4 ビットが S_n [3 : 0] である。この入力値と出力値とを示す真理値表を実現する組み合わせ論理回路を設計することで、5 倍回路を実現することができる。

【0055】

10

20

30

40

50

図6に戻り、上述のようにして実現された2倍回路及び5倍回路が、定数倍回路127に内蔵されている。これにより、定数倍回路127での定数倍処理を高速に実現することができる。またこのような定数倍処理の工夫だけでなく、図6の構成では、各減算器に対するN倍除数の割り当てについても工夫がされている。引き戻し法では各減算器のキャリーアウトによってN倍除数の選択を制御するので、各々の減算器に同時に入力するN倍除数の大小関係を常に同じにすることで、N倍除数の選択制御が簡易になる。前述したアルゴリズムの例でもそうして説明している。しかし図6の演算回路では、その関係を崩し、1回目の減算時には常に第1の減算器128へ5倍除数を入力するようになっている。これは5倍除数を使う時には常に第1の減算器128を用いることで、必要なセレクタの段数を減らすことができるためである。

10

【0056】

以下に、図6に示す演算回路の動作について説明する。なお以下の説明並びに図6及び以降の図において、中間剩余R、除数D I V s、N倍除数N D I V s、部分商Q [3:0]、減算回数判定信号c y c l e、商予測信号p r e Q、第1及び第2の減算器に出力するN倍除数×N a d d 1及び×N a d d 2と表記する。また更に、4倍除数選択信号s e 1 × 4、5倍除数選択信号s e 1 × 5、8倍除数選択信号s e 1 × 8、キャリーアウトC O 1及びC O 2、部分商Q 1及びQ 2、そして中間剩余の上位2桁R [7:0]、除数の上位1桁S [3:0]と表記する。減算回数判定信号c y c l eは、1回目の減算ループにおいて0であり、2回目の減算ループにおいて1となる。

【0057】

図6において、まず中間剩余レジスタ121と除数レジスタ122とに中間剩余Rと除数D I V sとが入力され、サイクルレジスタ123と4倍選択レジスタ124とに減算回数判定信号c y c l eと4倍除数選択信号s e 1 × 4とが入力される。中間剩余レジスタ121と除数レジスタ122とから、商予測回路125に、中間剩余Rと除数D I V sとが入力される。

20

【0058】

図9は、商予測回路125の構成の一例を示す図である。図9に示す商予測回路125は、AND回路141乃至151及びOR回路152乃至155を含む。AND回路141乃至144については、入力の一部が負論理となっている。この商予測回路125が、中間剩余の上位2桁R [7:0]と除数の上位1桁S [3:0]との情報に基づいて、この情報から予測可能な精度よりも低い精度での商予測を行う。即ち、商予測回路125が、この情報に基づいて商予測を行い、図2の表で区切り線10の上側のグループ（商が0～7）と区切り線10の下側のグループ（商が4～9）との何れか一方のグループを特定する。なお図9において、中間剩余の上位2桁R [7:0]の全ビットが使用されているのではない（例えばR [0]は使用されていない）。即ち、中間剩余の上位2桁R [7:0]と除数の上位1桁S [3:0]との全ビットではない一部のビットを用いて商予測を行っている。商予測回路125は、商の取りえる範囲が4～9であるならば1、0～7であるならば0となるようなセレクト信号を生成する。商予測回路125は、生成したセレクト信号を、倍数選択回路126、制御回路130、及び部分商計算回路131に供給する。

30

【0059】

図10は、定数倍回路127の構成の一例を示す図である。図10に示す定数倍回路127は、5倍回路161、2倍回路162乃至164、及びセレクタ165乃至167を含む。セレクタ166が選択して出力するN倍除数×N a d d 1が減算器128に供給される。セレクタ167が選択して出力するN倍除数×N a d d 2が減算器129に供給される。4倍除数選択信号s e 1 × 4、5倍除数選択信号s e 1 × 5、及び8倍除数選択信号s e 1 × 8は倍数選択回路126から供給される。

40

【0060】

定数倍回路127は、5倍除数選択信号s e 1 × 5が1であるならば5倍の除数を減算器128に供給し、5倍除数選択信号s e 1 × 5が0であるならばそのままの除数（1倍

50

の除数)を減算器 128 に供給する。定数倍回路 127 は、4 倍除数選択信号 $sel \times 4$ 及び 8 倍除数選択信号 $sel \times 8$ がそれぞれ 0 及び 0 であるならば、2 倍の除数を減算器 129 に供給する。定数倍回路 127 は、4 倍除数選択信号 $sel \times 4$ 及び 8 倍除数選択信号 $sel \times 8$ がそれぞれ 1 及び 0 であるならば、4 倍の除数を減算器 129 に供給する。定数倍回路 127 は、8 倍除数選択信号 $sel \times 8$ が 1 であるならば、8 倍の除数を減算器 129 に供給する。

【0061】

図 11 は、倍数選択回路 126 の構成の一例を示す図である。倍数選択回路 126 は、インバータ 171 及び AND 回路 172 を含む。AND 回路 172 の一方の入力は負論理 10 となっている。倍数選択回路 126 は、サイクルレジスタ 123 からの減算回数判定信号 $cycle$ 、4 倍選択レジスタ 124 からの 4 倍除数選択信号 $sel \times 4$ 、及び商予測回路 125 からの商予測信号 $preQ$ を、入力として受け取る。倍数選択回路 126 は、これらの入力に応じて、4 倍除数選択信号 $sel \times 4$ 、5 倍除数選択信号 $sel \times 5$ 、8 倍除数選択信号 $sel \times 8$ の値を 0 又は 1 に設定する。

【0062】

倍数選択回路 126 は、減算回数判定信号 $cycle$ が 0 であるならば 5 倍除数選択信号 $sel \times 5$ を 1 とする。倍数選択回路 126 は、4 倍除数選択信号 $sel \times 4$ については、受け取った信号をそのまま出力する。倍数選択回路 126 は、減算回数判定信号 $cycle$ が 0 であり且つ商予測信号 $preQ$ が 1 であるならば、8 倍除数選択信号 $sel \times 8$ を 1 とする。 20

【0063】

図 12 は、倍数選択回路 126 及び定数倍回路 127 の入出力の関係を示す表である。図 10 に示す定数倍回路 127 及び図 11 の倍数選択回路 126 は、図 12 に示す表に示されるように動作する。例えば、減算回数判定信号 $cycle$ 、4 倍除数選択信号 $sel \times 4$ 、及び商予測信号 $preQ$ が、それぞれ 0, 0, 0 である場合、5 倍除数選択信号 $sel \times 5$ 、4 倍除数選択信号 $sel \times 4$ 、及び 8 倍除数選択信号 $sel \times 8$ は、それぞれ 1, 0, 0 となる。またこの時、第 1 の減算器 128 (SUB1) に供給される N 倍除数は 5 倍除数であり、第 2 の減算器 129 (SUB2) に供給される N 倍除数は 2 倍除数である。 30

【0064】

図 6 に戻り、減算器 128 は、供給された中間剩余 R から N 倍除数を減算し、減算結果である中間剩余 R1 と、減算のキャリーアウト CO1 とを出力する。また減算器 129 は、供給された中間剩余 R から N 倍除数を減算し、減算結果である中間剩余 R2 と、減算のキャリーアウト CO2 とを出力する。部分商計算回路 131 に、キャリーアウト CO1 が供給される。部分商計算回路 131 に、キャリーアウト CO1 及び CO2 が供給される。また中間剩余選択回路 132 にも、キャリーアウト CO1 及び CO2 が供給される。 40

【0065】

図 13 は、中間剩余選択回路 132 の構成の一例を示す図である。図 13 に示す中間剩余選択回路 132 は、一部の入力が負論理の AND 回路 181 乃至 184、セレクタ 185 及び 186、及び OR 回路 187 を含む。中間剩余選択回路 132 は、減算回数判定信号 $cycle$ 、4 倍除数選択信号 $sel \times 4$ 、商予測信号 $preQ$ 、並びにキャリーアウト CO1 及び CO2 を入力として受け取る。 40

【0066】

図 14 は、中間剩余選択回路 132 の入出力の関係を示す表である。中間剩余選択回路 132 は、入力が与えられたときに、図 14 の表に示されるようなセレクト信号 $selR[1]$ 及び $selR[0]$ を出力する。セレクト信号 $selR[1]$ 及び $selR[0]$ は、図 6 に示されるように、セレクタ 134 に供給される。 50

【0067】

図 6 においてセレクタ 134 は、セレクト信号 $selR[1]$ 及び $selR[0]$ に応じて、中間剩余レジスタ 121 の中間剩余 R、減算器 128 の減算結果である中間剩余 R

1、又は減算器 129 の減算結果である中間剩余 R2 を選択する。選択された中間剩余は、中間剩余レジスタ 121 に供給され格納される。具体的には、セレクト信号 selR[1] 及び selR[0] がそれぞれ 0 及び 0 の場合、中間剩余 R が選択される。selR[1] 及び selR[0] がそれぞれ 0 及び 1 の場合、中間剩余 R1 が選択される。また selR[1] 及び selR[0] がそれぞれ 1 及び 0 の場合、中間剩余 R2 が選択される。

【0068】

図 15 は、制御回路 130 の構成の一例を示す図である。図 15 に示す制御回路 130 は、インバータ 191 と、入力の一部が負論理である AND 回路 192 とを含む。制御回路 130 は、減算回数判定信号 cycle、商予測信号 preQ、及びキャリーアウト CO1 を入力として受け取る。制御回路 130 は、減算回数判定信号 cycle を反転する。反転された減算回数判定信号 cycle は、サイクルレジスタ 123 に供給され格納される。また、減算回数判定信号 cycle が 0、商予測 preQ が 1、且つキャリーアウト CO1 が 0 である場合に限り、制御回路 130 は、4 倍除数選択信号 selx4 を 1 に設定する。1 に設定された 4 倍除数選択信号 selx4 は、4 倍選択レジスタ 124 に供給され格納される。

【0069】

図 16 は、部分商計算回路 131 の構成の一例を示す図である。図 16 に示す部分商計算回路 131 は、加算器 201、定数テーブル 202、AND 回路 203 乃至 205、及び OR 回路 206 を含む。AND 回路 204 及び 205 の一方の入力は負論理となっている。部分商計算回路 131 は、減算回数判定信号 cycle、4 倍除数選択信号 selx4、商予測信号 preQ、キャリーアウト CO1 及び CO2、及び部分商レジスタ 133 からの部分商 Q を入力として受け取る。

【0070】

図 17 は、部分商計算回路 131 の入出力の関係を示す表である。部分商計算回路 131 は、入力が与えられたときに、図 17 の表に示されるような部分商を出力する。なお図 17において、「行う処理」の Q として、4 や 5 等の数値が示されているときは、部分商 Q として当該数値を出力することを示す。また「行う処理」の Q として、+1 や +2 等の演算が示されているときは、現在の部分商 Q に対して当該演算を行うことを示す。

【0071】

図 18 は、図 16 の定数テーブル 202 の入出力の関係を示す表である。商予測信号 preQ、キャリーアウト CO1、及びキャリーアウト CO2 により、定数テーブル 202 中に格納される複数の定数のうちの 1 つが選択され、選択された定数が出力される。例えば、商予測信号 preQ、キャリーアウト CO1、及びキャリーアウト CO2 がそれぞれ 1, 1, 0 の場合、部分商 Q として 0101 が出力される。以下において、定数テーブル 202 の出力を第 1 の部分商と呼ぶ。

【0072】

図 16 を再び参照し、部分商レジスタ 133 からの部分商 Q とキャリーアウト CO1 とが加算器 201 に入力され、キャリーアウト CO2 が入力キャリーとして加算器 201 に输入される。この加算器 201 の加算結果を、以下において、第 2 の部分商と呼ぶ。

【0073】

4 倍除数選択信号 selx4 が 1 である時、部分商 Q が、OR 回路 206 を介して部分商計算回路 131 から出力される。出力された部分商 Q が、部分商レジスタ 133 に供給され格納される。減算回数判定信号 cycle が 0 である時、前述の第 1 の部分商が、OR 回路 206 を介して部分商計算回路 131 から出力される。出力された部分商 Q が、部分商レジスタ 133 に供給され格納される。また 4 倍除数選択信号 selx4 が 0 であり且つ減算回数判定信号 cycle が 1 である時、前述の第 2 の部分商が、OR 回路 206 を介して部分商計算回路 131 から出力される。出力された部分商 Q が、部分商レジスタ 133 に供給され格納される。

【0074】

10

20

30

40

50

以上のようにして、図6に示す演算回路119Aが動作することにより、図4に示すアルゴリズムが実行され、中間剰余と部分商とを2回の演算ループにて求めることができる。図6に示す演算回路119Aでは、単純な構成を有する商予測回路125により粗い商予測を行い、演算ループ回数を少なくする(図6の例では2回にする)ことができる。また減算ループ内における処理を単純な回路で実現することにより、高い動作周波数での実装が可能となり、小さな回路規模で高速に動作する演算回路を実現することができる。

【0075】

以上、本発明を実施例に基づいて説明したが、本発明は上記実施例に限定されるものではなく、特許請求の範囲に記載の範囲内で様々な変形が可能である。

【符号の説明】

10

【0076】

110 プロセッサ

111 メモリ

112 2次キャッシュ部

113 1次キャッシュ部

113A 命令キャッシュ

113B データキャッシュ

114 制御部

115 演算部

116 レジスタ

117 演算制御部

118 演算器

119 除算器

119A 演算回路

20

【図1】

【図2】

加算器の数と減算時に商が取りえる数との組み合わせに対し、
それぞれ残りの減算回数を示した表

		減算時に商が取りえる数									
		10	9	8	7	6	5	4	3	2	1
減 算 器 の 数	1	4	4	3	3	3	2	2	1		
	2	3	2	2	2	2	2	1	1	1	
	3	2	2	2	2	2	2	1	1	1	
	4	2	2	2	2	2	1	1	1	1	
	5										

中間剰余の上位2桁と除数の上位1桁との組み合わせテーブルを示す図

組合せ	01 xx	102 xx	103 xx	04 xx	05 xx	06 xx	07 xx	08 xx	09 xx
00_xx	0	0	0	0	0	0	0	0	0
01_xx	0	0	0	0	0	0	0	0	0
02_xx	01	0	0	0	0	0	0	0	0
03_xx	01	01	0	0	0	0	0	0	0
04_xx	21d	12	1	01	0	0	0	0	0
05_xx	21d	12	1	01	0	0	0	0	0
06_xx	34d	23	12	1	1	01	0	0	0
07_xx	34d	23	12	1	1	1	01	0	0
08_xx	34579	234	12	12	1	1	1	01	0
09_xx	45678	345	23	12	12	1	1	1	01
10_xx	45678	345	23	2	12	1	1	1	1
11_xx	56789	456	23	2	12	1	1	1	1
12_xx	56789	456	23	3	12	1	1	1	1
13_xx	67890	456	23	3	12	1	1	1	1
14_xx	67890	456	23	3	12	1	1	1	1
15_xx	78901	456	23	3	12	1	1	1	1
16_xx	89012	456	23	3	12	1	1	1	1
17_xx	90123	456	23	3	12	1	1	1	1
18_xx	91234	456	23	3	12	1	1	1	1
19_xx	92345	456	23	3	12	1	1	1	1
20_xx	93456	456	23	3	12	1	1	1	1
21_xx	94567	456	23	3	12	1	1	1	1
22_xx	95678	456	23	3	12	1	1	1	1
23_xx	96789	456	23	3	12	1	1	1	1
24_xx	97890	456	23	3	12	1	1	1	1
25_xx	98901	456	23	3	12	1	1	1	1
26_xx	99012	456	23	3	12	1	1	1	1
27_xx	90123	456	23	3	12	1	1	1	1
28_xx	91234	456	23	3	12	1	1	1	1
29_xx	92345	456	23	3	12	1	1	1	1
30_xx	93456	456	23	3	12	1	1	1	1
31_xx	94567	456	23	3	12	1	1	1	1
32_xx	95678	456	23	3	12	1	1	1	1
33_xx	96789	456	23	3	12	1	1	1	1
34_xx	97890	456	23	3	12	1	1	1	1
35_xx	98901	456	23	3	12	1	1	1	1
36_xx	99012	456	23	3	12	1	1	1	1
37_xx	90123	456	23	3	12	1	1	1	1
38_xx	91234	456	23	3	12	1	1	1	1
39_xx	92345	456	23	3	12	1	1	1	1
40_xx	93456	456	23	3	12	1	1	1	1
41_xx	94567	456	23	3	12	1	1	1	1
42_xx	95678	456	23	3	12	1	1	1	1
43_xx	96789	456	23	3	12	1	1	1	1
44_xx	97890	456	23	3	12	1	1	1	1
45_xx	98901	456	23	3	12	1	1	1	1
46_xx	99012	456	23	3	12	1	1	1	1
47_xx	90123	456	23	3	12	1	1	1	1
48_xx	91234	456	23	3	12	1	1	1	1
49_xx	92345	456	23	3	12	1	1	1	1
50_xx	93456	456	23	3	12	1	1	1	1
51_xx	94567	456	23	3	12	1	1	1	1
52_xx	95678	456	23	3	12	1	1	1	1
53_xx	96789	456	23	3	12	1	1	1	1
54_xx	97890	456	23	3	12	1	1	1	1
55_xx	98901	456	23	3	12	1	1	1	1
56_xx	99012	456	23	3	12	1	1	1	1
57_xx	90123	456	23	3	12	1	1	1	1
58_xx	91234	456	23	3	12	1	1	1	1
59_xx	92345	456	23	3	12	1	1	1	1
60_xx	93456	456	23	3	12	1	1	1	1
61_xx	94567	456	23	3	12	1	1	1	1
62_xx	95678	456	23	3	12	1	1	1	1
63_xx	96789	456	23	3	12	1	1	1	1
64_xx	97890	456	23	3	12	1	1	1	1
65_xx	98901	456	23	3	12	1	1	1	1
66_xx	99012	456	23	3	12	1	1	1	1
67_xx	90123	456	23	3	12	1	1	1	1
68_xx	91234	456	23	3	12	1	1	1	1
69_xx	92345	456	23	3	12	1	1	1	1
70_xx	93456	456	23	3	12	1	1	1	1
71_xx	94567	456	23	3	12	1	1	1	1
72_xx	95678	456	23	3	12	1	1	1	1
73_xx	96789	456	23	3	12	1	1	1	1
74_xx	97890	456	23	3	12	1	1	1	1
75_xx	98901	456	23	3	12	1	1	1	1
76_xx	99012	456	23	3	12	1	1	1	1
77_xx	90123	456	23	3	12	1	1	1	1
78_xx	91234	456	23	3	12	1	1	1	1
79_xx	92345	456	23	3	12	1	1	1	1
80_xx	93456	456	23	3	12	1	1	1	1
81_xx	94567	456	23	3	12	1	1	1	1
82_xx	95678	456	23	3	12	1	1	1	1
83_xx	96789	456	23	3	12	1	1	1	1
84_xx	97890	456	23	3	12	1	1	1	1
85_xx	98901	456	23	3	12	1	1	1	1
86_xx	99012	456	23	3	12	1	1	1	1
87_xx	90123	456	23	3	12	1	1	1	1
88_xx	91234	456	23	3	12	1	1	1	1
89_xx	92345	456	23	3	12	1	1	1	1
90_xx	93456	456	23	3	12	1	1	1	1
91_xx	94567	456	23	3	12	1	1	1	1
92_xx	95678	456	23	3	12	1	1	1	1
93_xx	96789	456	23	3	12	1	1	1	1
94_xx	97890	456	23	3	12	1	1	1	1
95_xx	98901	456	23	3	12	1	1	1	1
96_xx	99012	456	23	3	12	1	1	1	1
97_xx	90123	456	23	3	12	1	1	1	1
98_xx	91234	456	23	3	12	1	1	1	1
99_xx	92345	456	23	3	12	1	1	1	1
100_xx	93456	456	23	3	12	1	1	1	1
101_xx	94567	456	23	3	12	1	1	1	1
102_xx	95678	456	23	3	12	1	1	1	1
103_xx	96789	456	23	3	12	1	1	1	1
104_xx	97890	456	23	3	12	1	1	1	1
105_xx	98901	456	23	3	12	1	1	1	1
106_xx	99012	456	23	3	12	1	1	1	1
107_xx	90123	456	23	3	12	1	1	1	1
108_xx	91234	456	23	3	12	1	1	1	1
109_xx	92345	456	23	3	12	1	1	1	1
110_xx	93456	456	23	3	12	1	1	1	1
111_xx	94567	456	23	3	12	1	1	1	1
112_xx	95678	456	23	3	12	1	1	1	1
113_xx	96789	456	23	3	12	1	1	1	1
114_xx	97890	456	23	3	12	1	1	1	1
115_xx	98901	456	23						

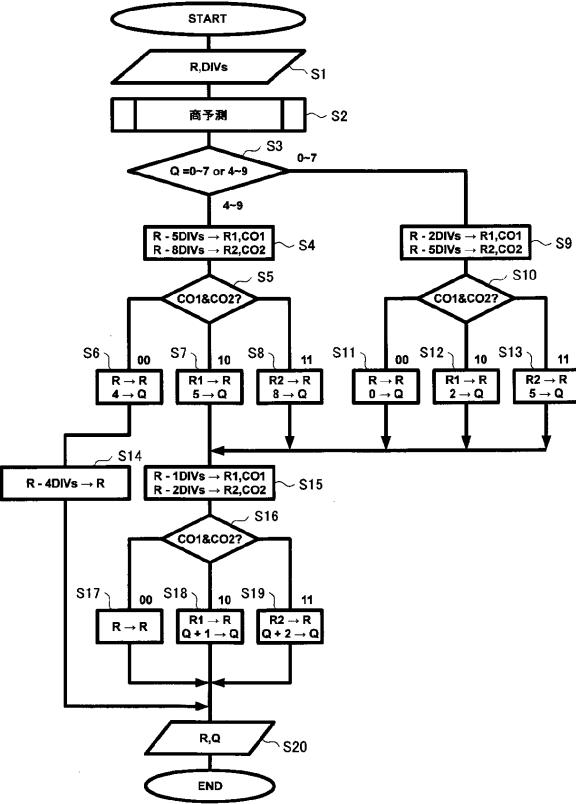
【図3】

2個の加算器により2回のループで結果を算出可能な演算器において、
1回目及び2回目の減算の結果の組み合わせに対して
結果として得られる部分商の値の一例を示した表

		1回目の減算		2回目の減算		選択された処理 R-10DIVs R-20DIVs R-40DIVs Q=+/-	商の範囲 R-10V / R-20V / R-40V	必要減算回数
並びえる 部分商の符号	選択される部分商	並びえる 部分商の符号	選択される部分商	並びえる 部分商の符号	選択される部分商			
負	負	4	R	4	-	*	*	*
負	正	-	-	-	-	正	R2	2
正	負	5	R1	57	-	負	R1	5
正	正	8	R2	89	-	正	R2	2
4~9								

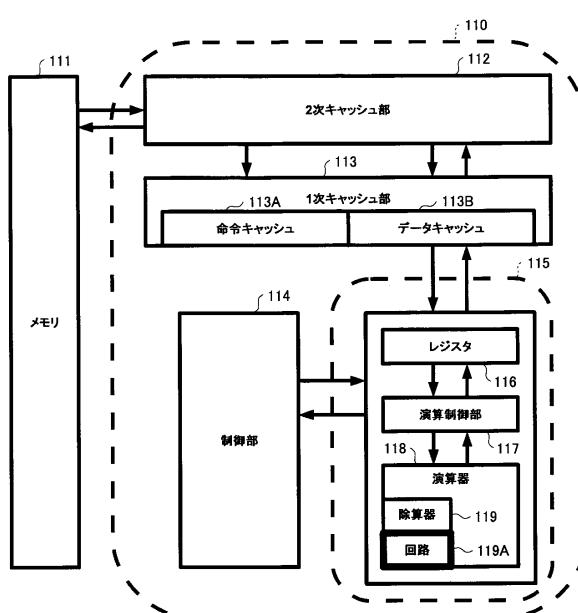
【図4】

図3に示したアルゴリズムの処理の流れを示すフローチャート



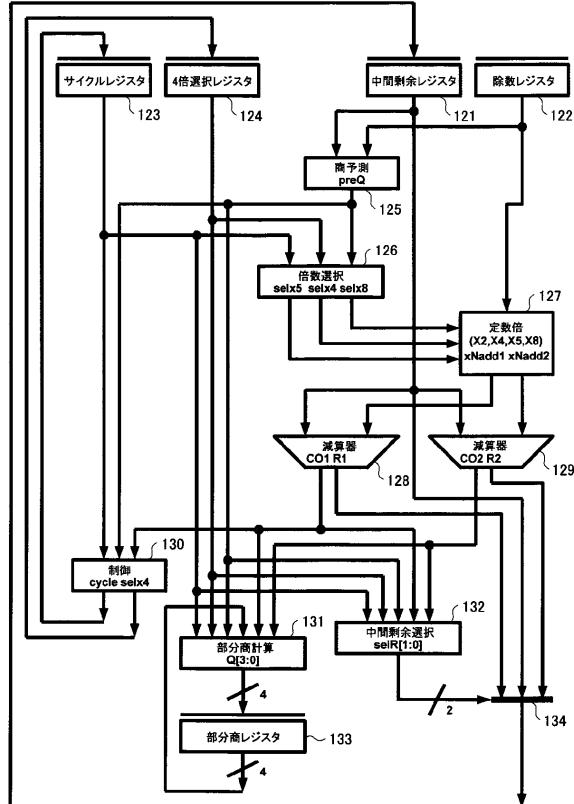
【図5】

コンピュータシステムの構成の一例を示す図



【図6】

演算回路の構成の一例を示す図



【図7】

2倍回路における各桁の入出力値を示す真理値表

$A_n[3:0]$	$S_{n+1}[0], S_n[3:1]$
0000	0000
0001	0001
0010	0010
0011	0011
0100	0100
0101	1000
0110	1001
0111	1010
1000	1011
1001	1100
1010	XXXX
1011	XXXX
1100	XXXX
1101	XXXX
1110	XXXX
1111	XXXX

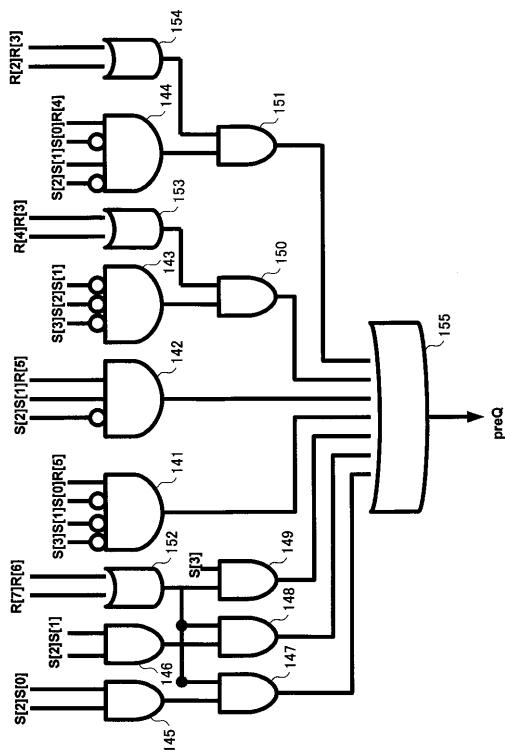
【図8】

5倍回路における各桁の入出力値を示す真理値表

$A_n[0]$	$A_{n-1}[3:1]$	$S_n[3:0]$
0	000	0000
	001	0001
	010	0010
	011	0011
	100	0100
	101	XXXX
	110	XXXX
	111	XXXX
1	000	0101
	001	0110
	010	0111
	011	1000
	100	1001
	101	XXXX
	110	XXXX
	111	XXXX

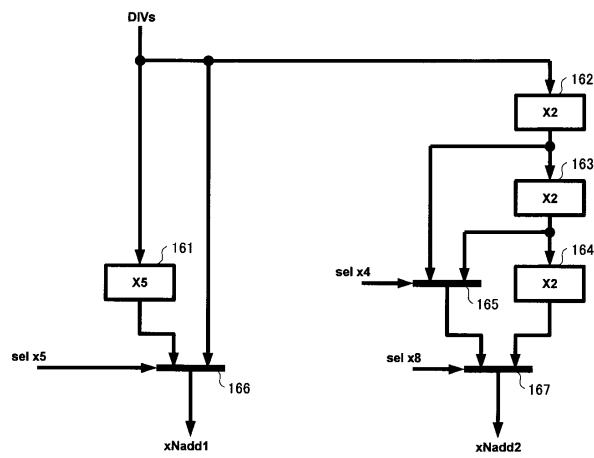
【図9】

商予測回路の構成の一例を示す図



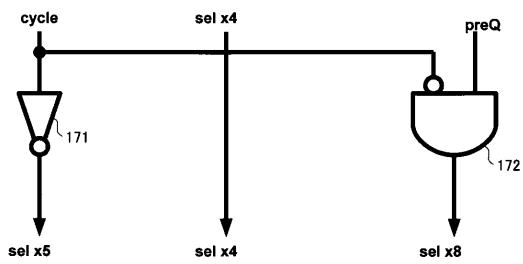
【図10】

定数倍回路の構成の一例を示す図



【図11】

倍数選択回路の構成の一例を示す図



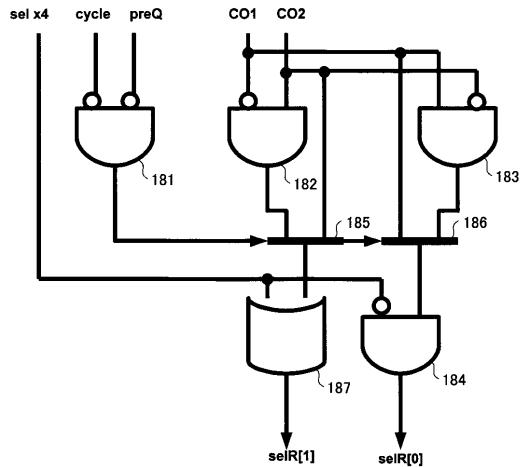
【図12】

倍数選択回路及び定数倍回路の入出力の関係を示す表

入力			セレクト信号			選択される倍数	
cycle	sel x4	preQ	selx5	selx4	selx8	SUB1	SUB2
0	X	1	1	—	1	5DIVs	8DIVs
		0		0	0		2DIVs
1	0	X	0	0	0	DIVs	2DIVs
		1	—	1	0	—	4DIVs

【図13】

中間剩余選択回路の構成の一例を示す図



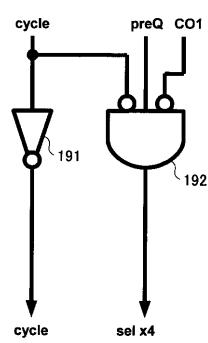
【図14】

中間剩余選択回路の入出力の関係を示す表

入力			セレクト信号			行う処理	
cycle	sel x4	preQ	CO1	CO2	selR[1]	selR[0]	R
0	0	1	0	0	0	0	R
			0	1	—	—	—
			1	0	0	1	R1
			1	1	1	0	R2
	0	X	0	0	0	0	R
			1	0	—	—	—
			0	1	1	0	R2
			1	1	0	1	R1
1	0	X	0	0	0	0	R
			0	1	—	—	—
			1	0	0	1	R1
			1	1	1	0	R2
	1	X	—	0	—	—	—
			1	1	1	0	R2

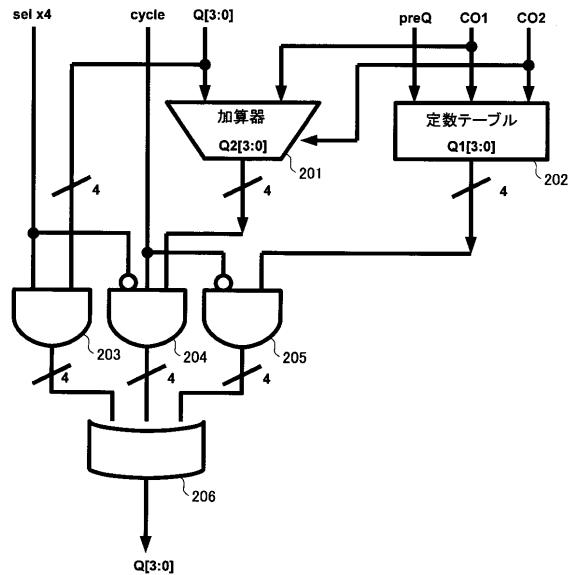
【図15】

制御回路の構成の一例を示す図



【図16】

部分商計算回路の構成の一例を示す図



【図17】

部分商計算回路の入出力の関係を示す表

入力				行う処理
<i>cycle</i>	<i>sel x4</i>	<i>preQ</i>	<i>CO1</i>	<i>Q</i>
0	X	1	0	0
			0	-
			1	5
			1	8
	0	0	0	0
			1	-
			0	2
			1	5
1	0	X	0	+0
			0	-
			1	+1
			1	+2
	1	X	-	-
			1	+0

【図18】

定数テーブルの入出力の関係を示す表

<i>preQ</i>	<i>CO1</i>	<i>CO2</i>	<i>Q[3:0]</i>
1	0	0	0100
	0	1	XXXX
	1	0	0101
	1	1	1000
0	0	0	0000
	1	0	XXXX
	0	1	0010
	1	1	0101

フロントページの続き

(72)発明者 北村 健一
神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

審査官 宮下 誠

(56)参考文献 特開昭58-114133(JP,A)
特開平07-160480(JP,A)

(58)調査した分野(Int.Cl., DB名)

G 06 F 7 / 535